

77-51038/495739 Построение индивидуального образовательного маршрута

Молодежный научно-технический вестник #11, ноябрь 2012

УДК: 004.415.25

автор: Большакова Д. О.

УДК 004.415.25

boldol@mail.ru

МГТУ им. Н.Э. Баумана

Введение

В современном мире человек вынужден постоянно осваивать новые технологии, получать знания в определенной предметной области, поэтому актуальным становится создание и применение систем дистанционного обучения (ДО). Основным преимуществом таких систем является возможность обучения в индивидуальном режиме. Удобство такой формы обучения привело к появлению ряда электронных курсов. Однако, обучаемый зачастую не способен самостоятельно сформировать траекторию обучения, включающую тот набор курсов, который необходим для достижения его учебных целей.[1] Поэтому возникла необходимость создания автоматизированных систем, содержащих средства построения индивидуальных образовательных маршрутов. Для решения этой задачи был предложен ряд методик, среди которых можно выделить следующие: [2]

- 1) Моделирование предметной области (*Domain Modeling*)
 - Граф образовательных путей (*Learning Path Graph*)
 - Карта понятий (*Concept Map*)
 - Онтология (*Ontology*)
 - Последовательность графов курсов образовательной деятельности (*Learning Activity Graph Curriculum Sequence*)
- 2) Управление выполнением граничных условий (*Handling Constraint Satisfaction*)
 - Генетическое программирование (*Genetic Programming*)
 - Метод роя частиц (*Swarm Intelligence*)
- 3) Группировка пользователей (*Clustering of Users*)

- Нейронные сети (*Neural Networks*)
- 4) Умозаключении, основанное на истории (*Case based reasoning*)
- 5) Учебный портрет предыдущего пользователя (*Learning Profile of Previous User*)
 - Колония муравьев (*Ant Colony*)
- 6) Статистический подход (*Statistical approach*)
 - Сеть Байеса (*Bayesian Network*)
- 7) Автоматическое формирование алгоритма поиска образовательного маршрута (*Automatic Generation of Learning Path Algorith*)
- 8) Взвешенный образовательный объект (*Weighted Learning Object*)
- 9) Подход, основанный на сети Петри (*Petri Nets Based Approach*)

Задачами данной работы являются поиск источников информации по проблеме построения индивидуальных траекторий обучения, нахождение существующих методов решения данной задачи, выявление основных подходов.

1. Обзор подходов к решению задачи

В гибких системах ДО при формировании индивидуального учебного контента должны быть учтены следующие основные факторы [2]:

- Уровень знаний обучаемого
- Темп обучения
- Уровень компетенций
- Цели обучения

Образовательный маршрут определяется как [2] последовательность образовательных показателей, которые характеризуют студента, выполняющего разделы курса в системе электронного обучения.

Под разделом (блоком) курса понимается [2] абстрактное представление курса, урока, семинара или любого другого формального или неформального события, обучающей направленности.

В рамках данной работы будут рассмотрены методы, указанные в статье [2]. Авторы данной работы [2] категоризируют методы решения задачи, а также приводят краткое описание каждого подхода.

1.1 Моделирование предметной области (*Domain Modeling*)

Моделирование предметной области является ключевой начальной точкой для реализации гибкого и адаптивного механизма в системе ДО.

1.1.1 Граф образовательных путей (*Learning Path Graph*)

LPG описывает структуру предметной области и соответствующие цели обучения.

Проектирование модели предметной области состоит из двух этапов [5]:

- Проектирование иерархии задач (целей)
- Составление набора понятий

LPG это направленный граф без циклов, который содержит все возможные образовательные маршруты, соответствующие выбранной цели. Чтобы построить *LPG* для каждого понятия из Графа понятий (*Concept Path Graph*) выбираются соответствующие объекты из пространства медиа-объектов.

Индивидуальный маршрут выбирается из графа, который состоит из всех возможных образовательных маршрутов и базируется на модели пользователя. Модель пользователя включает следующие характеристики обучаемого:

- Предпочтения
- Уровень компетенций
- Стиль обучения

LO (*learning object*) представляется в виде набора независимых параметров $g^{LO} = (g_1^{LO}, g_2^{LO}, \dots, g_n^{LO})$, а модель пользователя в виде $u^L = (u_1^L, u_2^L, \dots, u_m^L)$, тогда функцию годности можно представить в виде:

$$S^{L_j}(g^{LO_i}) = \frac{1}{n} \sum_{k=1}^n s_{g_k}^{L_j}(g_k^{LO_i})$$

Где следующая функция определяет степень значимости параметра g_k :

$$s_{g_k}^{L_j}(g_k^{LO_i}) = a_{g_k}^{L_j} + b_{g_k}^{L_j} g_k^{LO_i} \exp(-c_{g_k}^{L_j} g_k^{LO_i^2})$$

где $g_k^{LO_i}$ – значение LO_i ,

Параметры, определяющие форму функции:

$$a_{g_k}^{L_j} \in R, b_{g_k}^{L_j} \in R, c_{g_k}^{L_j} \in R$$

С учетом модели пользователя функция годности представляется в виде:

$$s_{g_k}^{L_j}(g_k^{LO_i}) = \begin{cases} s_{g_k}^{L_1}(g_k^{LO_i}), & \text{if } s_{g_k}^{L_1}(g_k^{LO_i}) = s_{g_k}^{L_2}(g_k^{LO_i}) \\ s_{g_k}^{L_1}(g_k^{LO_i}) + \frac{u_z^{L_j} - u_z^{L_1}}{u_z^{L_2} - u_z^{L_1}} [s_{g_k}^{L_2}(g_k^{LO_i}) - s_{g_k}^{L_1}(g_k^{LO_i})], & \text{if } s_{g_k}^{L_2}(g_k^{LO_i}) > s_{g_k}^{L_1}(g_k^{LO_i}) \end{cases}$$

Функция годности используется для задания веса для каждой связи в построенном *LPG* графе. Весовая функция представляется в следующем виде [5]:

$$W^{L_j}(g^{LO_i}) = 1 - S^{L_j}(g^{LO_i}) \in [0,1],$$

Затем во взвешенном графе выбирается наиболее подходящий обучаемому маршрут, посредством алгоритма нахождения кратчайшего пути (например, алгоритм Дейкстры).

1.1.2 Карта понятий (*Concept Map*)

Знания в предметной области представлены в виде Карты понятий. Используя графическое представление онтологий, вся структура курсов программного обучения и основные знания о предметной области могут быть четко выстроены. Карта понятий, базирующаяся на графических онтологиях, составляется на основе группировки курсов и корреляционной матрицы понятий. Для объединения связанных курсов в группу используются нечеткие схемы анализа кластеризации (*fuzzy clustering analysis schemes*). Карта понятий позволяет полностью описать структуру курса.

Вследствие алгоритма группировки (*clustering algorithm*) понятийно связанные учебные курсы будут объединены вместе, что способствует систематизации учебного контента. Это в свою очередь позволяет учащимся учиться систематически.

Таким образом, она заставляет обучающегося разбираться в отношениях между понятиями, предоставляя ему руководство по изучению.

1.1.3 Онтология (*Ontology*)

Онтологии используются для описания учебных материалов, которые компонуются для формирования учебного курса. Онтологии обеспечивают адаптивную электронную среду обучения и многократное использование образовательных ресурсов.

Онтология представляет собой набор абстрактных понятий, тем и семантические связи между ними, а не фактический материал обучения.

Понятие, которое характеризует ряд знаний помещается в некоторый класс «Понятие». Этот класс содержит данные типа «Наименование понятия», чтобы идентифицировать понятие, и другие свойства, которые позволяют устанавливать связи между понятиями предметной области.

Система электронного обучения выбирает наиболее подходящий для учащегося набор обучающих материалов.

1.1.4 Последовательность графов курсов образовательной деятельности (*Learning Activity Graph Curriculum Sequence*)

Под учебной деятельностью (*Learning Activity*) понимается любой объект от простого упражнения до сложного курса.

Граф учебной деятельности (*Learning Activity Graph*) используется для организации учебных ресурсов в учебные задачи. Учебная деятельность представлена как направленный граф с состояниями до и после в каждой вершине. Вершины графа могут представлять собой как простые объекты учебной деятельности, так и подграфы, включающие ряд взаимосвязанных учебных ресурсов.

Алгоритм построения последовательности учебной деятельности (*Learning activity sequencing algorithm*) объединяет использование модели учащегося и графа учебной деятельности для выстраивания последовательности прохождения объектов учебной деятельности. Этот подход может вырабатывать эффективные последовательности объектов учебной деятельности для различных учащихся, а также фиксировать учебный процесс и достижения для обновления модели пользователя.

1.2 Управление выполнением граничных условий (*Handling Constraint Satisfaction*)

Последовательность обучающих модулей определяется в терминах компетенций таким образом, что проблема выстраивания последовательности может быть рассмотрена как классическая задача с ограничениями (*Constraint Satisfaction Problem*). При этом, ограничения должны быть определены как требования, предъявляемые к уровню знаний до и после прохождения курса. Для построения адаптивного обучающего пути, или решения задачи с ограничениями применяются генетическое программирование и метод роя частиц (*Swarm Intelligence*).

1.2.1 Генетическое программирование (*Genetic Programming*)

Для построения обучающего пути в этом подходе выбираются разные типы учебных программ. Каждому учебному плану присваивается серийный номер. Начальный размер популяции фиксируется как 50. Функция пригодности позволяет судить о качестве сгенерированного обучающего пути. При определении функции пригодности учитываются результаты предварительного тестирования обучаемого, сложность учебного плана и уровни взаимосвязей между учебными планами. [3]

Операции репродукции, мутации осуществляются для нахождения следующего поколения обучающих путей. Каждое поколение путей обучения оценивается с помощью функции пригодности, а затем выбираются наиболее подходящие маршруты.

1.2.2 Метод роя частиц (*Swarm Intelligence*)

Это один из методов искусственного интеллекта, основанный на коллективном поведении децентрализованной, самоорганизующейся системы. Метод роя частиц (*PSO*) является эволюционным алгоритмом оптимизации. *PSO* имитирует поведение насекомых, таких как пчелы, например. Случайно проинициализированные частицы популяции летят через пространство решений, распространяя информацию, которую они собрали. Частицы используют эту информацию для динамического регулирования их скорости и взаимодействуют для поиска решения.

1.3 Группировка пользователей (*Clustering of Users*)

В этом подходе вместо создания одного пути для каждого ученика, учащиеся группируются в кластеры различных стилей обучения и обучающий маршрут подбирается для такого кластера.

1.3.1 Нейронные сети (*Neural Networks*)

В этой методике используется интеллектуальный агент, называемый «помощник в обучении». Функциональные возможности помощника бывают двух следующих типов:

- Обучение нейронной сети путем классификации каждого ученика в различные кластеры учащихся на основе их уровня знаний, предпочтений учащихся и темпов обучения.
- Создание обучающего пути, который подходит для кластера, в который ученик был классифицирован.

Самоорганизующейся карты (*SOM*) это нейронная сеть, которая используется для группировки сходных учеников.

1.4 Умозаключении, основанное на истории (*Case based reasoning*)

Умозаключении, основанное на истории (*CBR*) означает адаптацию старых решений для удовлетворения новым требованиям, с использованием старых методик, для объяснения новых ситуаций, или исходя из прецедентов для интерпретации новых ситуаций. Планировщик *CBR* должен быть системой дистанционного обучения, потому что он должен повторно использовать его собственный опыт. Этот метод

используется для нахождения индивидуального пути в случае если обучающийся уже проходил по аналогичному маршруту и не достиг поставленной цели.[2,3]

1.5 Учебный портрет предыдущего пользователя (*Learning Profile of Previous User*)

Такой подход генерирует обучающий маршрут принимая во внимание профиль обучения предыдущего пользователя с помощью расширенного системного подхода метода колонии муравьев.

1.5.1 Колония муравьев (*Ant Colony*)

Метод оптимизации колонии муравьев используется для прогнозирования наилучшего образовательного маршрута для студента с учетом профиля пользователя и путей, пройденных студентом ранее в системе электронного обучения.

Система колонии муравьев, базирующаяся на атрибутах (*Attribute-based Ant Colony System*) — это расширение системы колонии муравьев. Это метод нахождения учебных объектов подходящих для обучаемого на основе наиболее часто проходимых ранее другими пользователями маршрутов. Система обновляет пути исходя из уровня знаний обучаемого и стиля обучения пользователя. Это позволяет создать механизм поиска подходящих учебных объектов.

Есть три предпосылки для достижения этой цели:

- Адаптивный портал обучения содержит атрибуты учащегося, которые включают в себя уровень знаний учащегося и стиль обучения.
- Атрибуты учащегося и атрибуты учебных объектов, которые были занесены в систему учителем или контент-провайдеров
- Соответствующие отношения между учащимися и обучающими объектами.

1.6 Статистический подход (*Statistical approach*)

В рамках этого подхода, для нахождения адаптивного обучающего пути используются статистические методы, такие как, например, теория вероятностей Байеса (*Bayesian probability theory*).

1.6.1 Сеть Байеса (*Bayesian Network*)

Сеть Байеса это ориентированный граф, узлы которого представляют собой неопределенные искомые переменные, а ребра являются случайными связями между переменными. Каждому узлу ставится в соответствие значение в таблице вероятности узлов. Таблица содержит условные вероятностные значения.

Нахождение обучающего пути по помощи сети Байеса включает в себя два этапа. На первом шаге по байесовской теории вероятности создается таблица вероятностей. Таблица вероятности указывает на вероятность в различных узлах, которые могут быть достигнуты из текущего узла. Вероятностное значение задается в зависимости от уровня знаний, стиля обучения и темпа обучения ученика и называется “путь обучения учащегося” (*candidate learning paths*). Второй шаг состоит в построении Байесовской сети, расчета значения вероятностей для каждой единицы знаний в обучающем пути. Исходя из этого выбирается подходящий кратчайший путь.[4]

1.7 Автоматическое формирование алгоритма поиска образовательного маршрута (*Automatic Generation of Learning Path Algorithm*)

Такой подход способствует активному обучению, позволяя строить курсы, которые являются персонализированными с точки зрения как содержания, так и учебных материалов.

Персонализация осуществляется на основе профиля каждого студента, в котором отображается уровень знаний, предпочтения и цели учащегося. Таким образом, студентам предоставляется адаптивная среда, которая динамически регулирует содержание курсов во время процесса обучения. Алгоритм позволяет студенту принять участие в онлайн курсах. Каждый студент регистрируется в системе и присоединяется к какой-либо группе пользователей. Затем он выбирает курс и, следовательно, все связанные пути обучения строятся на основе указанных потребностей студента. Наконец, студент выбирает один из этих путей.

1.8 Взвешенный образовательный объект (*Weighted Learning Object*)

Такой подход включает в себя три основных этапа для создания адаптивного содержания курса.

Первый этап это оценка процесса обучения студента. Второй этап заключается в выборе курса, содержимое которого подходит для каждого ученика исходя из модели учащегося. Материал курса, текст, изображения, видео хранятся в базе данных учебных объектов и дополняют некоторые атрибуты для каждого учебного предмета, например, уровень сложности учебного объекта, общее время, необходимое для изучения и т.д. Адаптивная генерация содержания курса происходит в зависимости от уровня знаний обучаемого, предпочтений и целей учащегося. Последний этап заключается в выборе пути для каждого ученика.

1.9 Подход, основанный на сети Петри (*Petri Nets Based Approach*)

Сеть Петри состоит из двух видов узлов, называемых места и переходы, которые соединяются дугами. Для описания поведения системы, время связывается с такими атрибутами как место (Р-время), жетоны (так называемый возраст) и переходы (Т-время).

2. Генетическое программирование (*Genetic Programming*)

GP является одним из самых популярных методов решения задачи построения индивидуальных траекторий обучения. Применение генетического алгоритма к данной задаче подробно описывается в статье [3] приведенного списка литературы.

2.1 Пример использования алгоритма

Большинство создаваемых сейчас систем дистанционного обучения ориентированы на профиль и предпочтения ученика, и индивидуальный маршрут строится именно на основе этих параметров. Однако при построении системы обучения немало важно учитывать насколько сложность курса отвечает способностям ученика. Для этого в статье [3] предлагается использовать метод тестирования. (*CAT*)

В целом в литературе выделяются две модели обучения: традиционная и усовершенствованная. Усовершенствованная модель состоит из 4 этапов:

- 1) Выделение учителем базовых идей, разделов курса (*unit*).
- 2) Тестирование А
 - Расширение курса
 - Корректирующая деятельность и повторное тестирование В
- 3) Переход к следующему разделу.

Как правило, раздел (*unit*) состоит из ряда глав. Если ученику не удастся пройти тестирование А, для него посредством *GA* генерируется новая последовательность глав раздела.

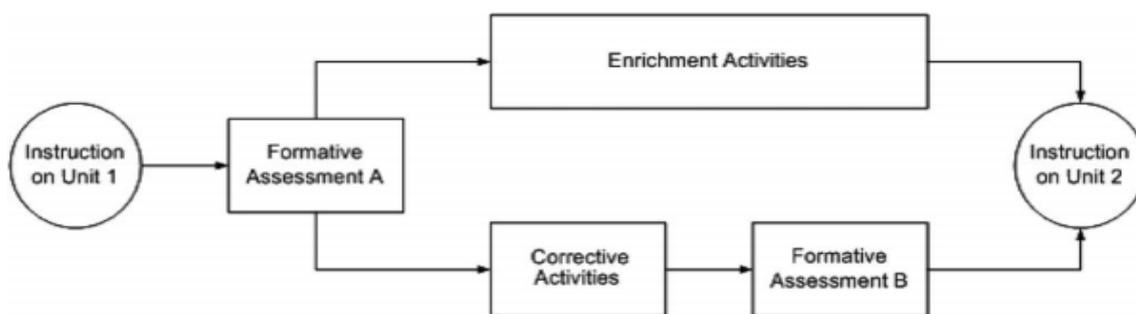


Рис.1. Архитектура системы

2.2 Описание метода

GA, как правило, используется для нахождения глобального оптимума функции.

Процесс оптимизации включает в себя следующие стадии:

- Инициализация (*initialization*)
- Селекция (*selection*)
- Скрещивание (*crossover*)
- Мутация (*mutation*)

Этапы генерации обучающего маршрута:

1) Определение хромосомной строки.

Порядковый номер присваивается каждому учебному курсу от 1 до n , где n – общее количество курсов, доступных учащемуся. Таким образом, порядковый номер каждого учебного курса соответствует порядковому номеру набора учебных программ составляющий общий курс (т.е. общий курс состоит из набора учебных курсов) или сгенерированный индивидуальный путь для генетического алгоритма.

2) Начальный размер популяции.

Как правило, начальный размер популяции может быть определен в зависимости от сложности решаемой задачи. Большая численность популяции приведет к сокращению скорости поиска GA, но это увеличит вероятность нахождения качественного решения. Для решения поставленной задачи размер принимается равным 50.

3) Выбор функции пригодности (*fitness function*).

Функция позволяет судить о качестве сгенерированного индивидуального учебного пути. Для примера, описанного ранее (2.1) в качестве исходной хромосомной строки рассматривается только набор курсов, для которых учащийся дал неправильные ответы на этапе тестирования A. Более того, курсу с наименьшей трудностью присваивается наименьший порядковый номер. Таким образом, функции формируется следующим образом:

$$f = \sum_{i=2}^n (w \times r_{(i-1)i} + (1 - w) \times d_i), \text{ где}$$

$w = f$ (затраченное время \требуемое время),

f – функция пригодности,

$r_{(i-1)i}$ – степень родства понятий $(i - 1)$ и i курса,

d_i – степень сложности курса,

w – уровень обучения,

n – общее число курсов.

- Операция репродукции
- Скрещивание
- Мутация
- Критерий останова

Генетический алгоритм неоднократно запускает операции скрещивания, мутации до выполнения критерия останова. Критерий для данной задачи установлен в 100 поколений.

3. Умозаключение, основанное на истории (*Case based reasoning*)

При рассмотрении данного метода положим за основу систему ДО, описанную ранее в данной статье и приведенную в статье [3]. *CBR* позволяет адаптировать старые решения в соответствии с новыми требованиями. Фактически, это метод машинного обучения. *CBR* требует наличия системы, которая обладает информацией что и когда должно быть изучено. Система обучения должна быть разделена на 3 типа:

- Планы обучения (*learning plans*)
- Ожидания от обучения (*learning expectations*)
- Критика обучения (*learning criticisms*)

В общем случае процедура логических рассуждений в *CRB* системах состоит из 5 шагов:

- Назначение указателей (*Assigning indexes*)
- Поиск случая (*Case retrieval*)
- Адаптация случая (*Case adaptation*)
- Тестирование случая (*Case testing*)
- Сохранение случая (*Case storage*)

Для системы, описанной ранее каждая индивидуальная последовательность курсов и результаты тестирования А сохраняются в системе. Таким образом, *CRB* может предоставить ученику итоговый тест, после прохождения учеником нескольких курсов и ряда тестов к ним.

3.1 Процесс обучения в *CRB* подсистеме

Есть два типа тестов: итоговые и промежуточные. Итоговые тесты используются для оценки эффективности обучения.

CRB применяется для анализа результатов итогового теста. Кроме того, *CRB* также предоставляет возможность формирования корректирующих мероприятий, в том числе различных материалов курса и промежуточного тестирования В.

1) Ввод нового случая

Новый случай означает, что учащийся не достиг надлежащего уровня в текущем разделе курса (*unit*). Таким образом, новый случай переключит систему *CBR* для предсказания вероятности конкретного случая. Механизм умозаключения *CRB* начнет искать случай наиболее близкий к новому случаю в целях поддержки корректирующей деятельности. В то же время учащийся может изучать различные материалы той же тематики. После этого он также может пройти тест, организованный системой *CBR*.

2) Анализ запроса

Случай состоит из многих указателей. Выбор какого-либо указателя как наиболее важного фактора оценки случая значительно влияет на полноту дела и его результаты оценки.

3) Назначение веса индекса

На основе различных уровней значимости между указателями оценки результатов, ожидается, что оценка результата для нового дела будет более подходящей. В рассматриваемом исследовании, система может задавать вес в соответствии с критериями, заданными экспертами.

4) Поиск случая

CRB сохраняет в системе данные о случаях. Он находит решение наиболее похожего дела путем сравнения похожих характеристик. Нахождение такого аналогичного случая одна из важнейших задач *CRB*.

Поиск случая обычно попадает в одну из четырех категорий: ближайший сосед, индуктивное обучения, учебное руководство, и любые их комбинации. *CRB* в рассматриваемом случае сочетает учебное руководство метода весовых коэффициентов (*WRF*). Метод *WRF* является производным от метода ближайшего соседа. Вес тривиальных функций принимается равным "0". Методом учебного руководства новый случай относится к сходной по свойствам группе. Далее, методом *WRF* используется для получения наиболее сходного случая из набора случаев библиотеки. Общее сходство определяется путем *WRF* метода, где функция определения сходства представлена следующим образом:

$$\text{Сходство} = \frac{\sum_{i=1}^n W_i * Z_i}{\sum_{i=1}^n W_i} \text{ если } X_i < Y_i, \text{ тогда } Z_i = \frac{X_i}{Y_i} \text{ иначе } Z_i = \frac{Y_i}{X_i}$$

W_i – вес свойства i ,

X_i – значение свойства в новом случае,

Y_i - значение свойства для случая в библиотеке.

5) Адаптация и повторное использование случая

Целью адаптации библиотечного случая является изменение данного случая для решения проблемы нового случая.

Могут быть применены следующие четыре метода адаптации: точная адаптация, интерполяция, использование средних значений для адаптации, и применение правил адаптации.

Точная адаптация требует эксперимент, который точно описывает разницу между двумя значениями параметра в библиотечном и новом случае, соответственно. Интерполяция осуществляется линейно и взвешивается отношением опоры и общим диапазоном параметров. Другим вариантом является использование средних значений. Если значение параметра v_1 должно быть адаптировано к значению v_2 , метод средних значений собирает случаи, которые содержат либо значения v_1 , либо значения v_2 . Система рассчитывает средние значения параметров, в рамках этих двух групп. Отношение второго значения к первому используется в качестве правила адаптации. Окончательный метод адаптации применяет правила адаптации. Правила адаптации описывают изменения, которые выражены факторами, различиями.

Тем не менее, основной функциональной возможностью вышеупомянутых четырех методов адаптации является предопределение правил адаптации для системы.

1) Пересмотр случая

Когда решения аналогичных случаев, которые были выбраны, не подходят для нового случая, изменения могут быть проведены на основе профессиональных знаний чтобы сделать их более уместными в данном случае с целью определения окончательных результатов.

2) Сохранение случая.

Сохранение случая в библиотеке случаев для повышения ее полноты является основным механизмом системы.

3) Аналитическое представление случая

Когда учащиеся проходят тест А, запись сохраняется в персональной базе данных. Запись это историческая информация, в том числе общий балл, правильный ответ и неправильный ответ. Затем новый случай вызовет переключение системы *CRB* для прогнозирования вероятности для данного конкретного случая. Система составляет таблицы случаев, которые делятся на таблицы правильных и неправильных вариантов. Правильные случаи используются для сравнения уровня знаний учащихся с некоторым

требуемым уровнем. Неправильные случаи используются для исправления ошибок учащихся.

4. Сеть Байеса (*Bayesian Network*)

Применение метода к задаче поиска индивидуального маршрута описано в статье [4] приведенного списка литературы.

Параметры учебных объектов, а также сами объекты определены в спецификации стандартов метаданных для систем электронного обучения. В приведенном в статье методе данные учебные объекты (*LO*) дополняются рядом атрибутов. В связи с этим вся модель системы делится на 3 модуля:

- Модуль учащегося (моделирование профиля ученика)
- Модуль содержания (генерация подходящего для ученика образовательного маршрута)
- Модуль представления (построение структуры курса)

Кроме того, для обеспечения гибкости системы раздел курса (*knowledge unit*) соответствует стандарту *SCORM* и дополняется рядом атрибутов:

- Предпосылки
- Уровень образования
- Уровень сложности
- Требуемое время
- Отношения между учебными объектами
- Стилль обучения
- Навыки

Так как раздел курса может состоять из одного или нескольких учебных объектов, учебные объекты также характеризуются приведенным набором атрибутов.

Процесс генерации образовательного маршрута включает следующие этапы:

1) Оценка ученика. Данный процесс базируется на профиле ученика и его запросах. Он предназначен для выделения ключевых требований к выбранному учеником курсу и классификации ученика.

2) Построение карты знаний (*knowledge mapping*). Преподаватель или автор курса составляет карту знаний. Она представляется в виде графа, где вершинами являются разделы курса, а дугами взаимосвязь между разделами.

3) Выбор образовательного пути. Основываясь на *LO*, профиле ученика на данном шаге выбирается набор индивидуальных путей. Каждый такой путь соответствует одному атрибуту *LO*. Таким образом, данный атрибут является ключевым для разделов курса. В

этой связи, каждому такому атрибуту соответствует индивидуальный путь. Пути являются независимыми друг от друга.

4) Генерация образовательного маршрута. Для выбора необходимого пути, из найденных на предыдущем этапе конструируется сеть Байеса.

Под картой знаний тут понимается направленный граф $G = (V, E)$, в котором

$V = \{v_0, v_1, \dots, v_n\}$ – набор вершин - разделы курса

$E = \{e_0, e_1, \dots, e_n\}$ - набор дуг - взаимосвязи между разделами

Каждая дуга имеет вес w , иллюстрирующий сложность достижимости одной вершины из другой.

Индивидуальный путь – набор вершин (разделов), которые должны быть пройдены в курсе. Где v_s – начальная вершина, v_e – конечная (целевая) вершина.

Образовательный маршрут – это индивидуальный путь, для которого сумма весов дуг максимальна или минимальна.

Для поиска индивидуального пути в статье предлагается алгоритм поиска кратчайшего пути в графе:

Input: Карта знаний $G = \{V, E\}$; v_s – начальная вершина, v_e – конечная (целевая) вершина.

Output: индивидуальный путь

Begin

```

Begin
  S = {Vs}
  For i := 2 to n do
    Begin
      D[i] := C[1, i];
      P[i] = {Vs};
    End;
  While V - S = ∅ do
    Begin
      Select v ∈ V - S that D[v] → min
      S := S ∪ {v};
      For each w ∈ V - S do
        If D[v] + C[v, w] < D[w] then
          Begin
            D[w] := D[v] + C[v, w];
            P[w] := v;
          End;
        End;
      End;
    End;
  End;

```

Рис.2 Алгоритм поиска

$C[i, j]$ - вес дуги между вершинами i и j . Если i и j не связаны, то $w = \infty$.

$D[u]$ – отношение между $\{V_s\}$ и u

$P[u]$ шаг пути, с $P[u] = v$ если есть путь $v \rightarrow u$.

Поиск образовательного маршрута осуществляется путем построения сети Байеса (BBN). BBN – направленные граф с вершинами (искомыми переменными) и дугами –

(независимыми связями между параметрами). С каждым узлом ассоциирована ячейка в таблице вероятности.

Поиск образовательного маршрута включает в себя 2 стадии:

1) Построение таблицы вероятности разделов курса для графа.

На предыдущем шаге был найден набор индивидуальных путей, отвечающих атрибутам LO . Теперь необходимо построить таблицу вероятности, которая проиллюстрирует общую вероятность появления вершин (независимых разделов) в образовательном маршруте. Для начальной и конечной вершин вероятность устанавливается равной 1. Для промежуточных она подсчитывается как отношение появления вершины в индивидуальном пути к общему числу индивидуальных путей.

2) Построение BBN для подсчета вероятностного значения для каждой вершины графа.

На данной стадии необходимо представить карту знаний в виде BBN .

$X = \{X_1, X_2, \dots, X_n\}$ – набор параметров, определяющих структуру курса.

P – условная вероятность.

В данном случае X – набор вершин графа.

X_i – текущая вершина

Pa_i – родительские вершины X_i

Для каждой вершины вычисляется условная вероятность. При этом учитывается достижимость текущей вершины из других вершин графа (родительских)

$$p(x) = \text{Pr}(X_i | Pa_i).$$

Таким образом, каждой вершине соответствует вероятностное значение ее вхождения в образовательный маршрут. Вероятность начальной и конечной вершин равна 1. Образовательный маршрут состоит из вершин с наибольшим значением вероятности.

Заключение

Соединение информационных технологий и инновационных педагогических методик позволяет повысить качество учебного процесса, поскольку дает преподавателям дополнительные возможности для построения индивидуальных образовательных траекторий путем реализации дифференцированного подхода к учащимся с разным уровнем готовности к обучению.

Для построения траектории индивидуального обучения в рассмотренных мной источниках предлагается использовать ряд подходов, основные из которых были кратко рассмотрены в данной работе. Безусловно, каждый подход имеет ряд преимуществ и недостатков. Основные из которых упомянуты в статье [3].

Подводя итог работы, стоит отметить, что наиболее популярными подходами являются ГА и нейронные сети. Кроме того, данная задача является актуальной и единого, стандартизированного подхода для ее решения на сегодняшний день не существует.

Литература

1. А.В.Кучер, В.В. Сокол, Н.С. Лесная, А.В. Бочаров “Архитектура системы построения индивидуальной траектории обучения, базирующейся на образовательном стандарте.” Вестник ХНТУ №2(38), 2010г., Проблемы высшей школы
2. Manju Bhaskar, Minu M Das “Genetic Algorithm Based Adaptive Learning Scheme Generation For Context Aware E-Learning”, (IJCSE) International Journal on Computer Science and Engineering Vol. 02, No. 04, 2010, 1271-1279
3. Mu-Jung Huang, Hwa-shan Huang, Mu-Yen Chen. “Constructing a personalized e-learning system based on genetic algorithm and casebased reasoning approach.” Expert Systems with Applications 33 (2007) 551-564
4. Nguyen viet anh, Nguyen viet ha, Ho si dam. “Constructinga Bayesian Belief Network to Generate Learning Path in Adaptive Hypermedia System.” T.24, S.1 (2008).
5. Pythagoras Karampiperis, Demetrios Sampson.” Adaptive Learning Resources Sequencing in Educational Hypermedia Systems.” Educational Technologies and society, 8(40), 128-147.