

Реализация методов адаптивного обучения

И. А. Кречетов, В. В. Романенко

Статья поступила
в редакцию
в декабре 2019 г.

Кречетов Иван Анатольевич
заведующий лабораторией инструментальных систем моделирования и обучения Томского государственного университета систем управления и радиоэлектроники. E-mail: kia@2i.tusur.ru

Романенко Владимир Васильевич
кандидат технических наук, доцент кафедры автоматизированных систем управления Томского государственного университета систем управления и радиоэлектроники. E-mail: rva@2i.tusur.ru

Адрес: 634034, Томск, ул. Красноармейская, 146.

Аннотация. Идеям адаптивного обучения уже не один десяток лет, но до сих пор большинство теоретических работ не получило практического воплощения, а программные решения долгое время не имели значимого распространения из-за недостаточного развития технологий электронного обучения. Современный уровень информационных технологий позволяет реализовывать сложные задачи в области обработки больших данных и искусственного интеллекта, к которой и относится адаптивное обучение.

В статье рассматривается практическая реализация методов и технологий адаптивного обучения. Предлагаемые решения позволяют строить

оптимальную траекторию изучения студентом модулей электронного курса. Критерием оптимальности является достижение максимального уровня знаний на момент окончания курса при минимальном времени обучения. Для решения данной оптимизационной задачи используется генетический алгоритм. Для экстраполяции уровня остаточных знаний студента была применена модель, основанная на скорости забывания информации.

Практическая реализация предлагаемой технологии включает набор инструментальных средств, расширяющих возможности системы дистанционного обучения по работе с адаптивными обучающими курсами, а также модуль, обеспечивающий работу генетического алгоритма. Разработаны несколько вариантов архитектуры программного решения с использованием разных технологий и языков программирования, а также с применением одного или двух серверов. Выполнена апробация решения при разработке адаптивных обучающих курсов для НИТУ «МИСиС» и ТУСУР.

Ключевые слова: адаптивное обучение, электронное обучение, генетический алгоритм, система дистанционного обучения.

DOI: 10.17323/1814-9545-2020-2-252-277

Массовые открытые онлайн-курсы (МООК) изменили подход к фундаментальному образованию, стерли границы времени и места обучения и в настоящее время продолжают трансформировать контексты своих применений. Современные тренды

электронного обучения призваны не заменить МООК новыми решениями, а скорее повысить эффективность приобретения знаний студентами и их вовлечение в образовательный процесс. Одним из таких трендов является адаптивное обучение.

Адаптивное обучение учитывает предыдущий учебный опыт студента, отслеживает текущий процесс получения знаний и качество их усвоения. В адаптивном обучении каждый студент проходит свой уникальный маршрут, который подстраивается под задачи обучения и способность обучаемого воспринимать и обрабатывать информацию.

Среди исследователей адаптивного обучения можно выделить П. Брусилковского [Brusilovsky, 1996; 1997; 1998; 2001], И. П. Норенкова [Норенков, Уваров, 2005; Норенков, Соколов, 2009; Норенков, Соколов, Уваров, 2009], Г. В. Рыбину [2008а; 2008б; 2010; 2011; 2014], Л. А. Растригина [Растригин, Эренштейн, 1988]. Труды этих авторов лежат в основе современных исследований данной темы, а также существующих программных решений. Однако долгое время технологии адаптивного обучения не имели значимого распространения по ряду причин:

- недостаточный уровень производительности вычислительной техники;
- состояние науки в данной области;
- сложности программной реализации интеллектуальных и экспертных систем;
- недостаточная развитость и распространенность технологий электронного обучения.

Современный уровень развития информационных технологий позволяет относительно незатратно реализовывать сложные задачи в области систем искусственного интеллекта и обработки больших данных, которые стали наиболее актуальными в период широкого распространения электронного обучения и сформировали новую нишу программных продуктов в сфере образования. К их числу относятся сервисы и системы адаптивного обучения.

Элементы адаптивного обучения сегодня применяют многие компании, выпускающие цифровые продукты. Степень и методы адаптации при этом могут быть разными, так как они во многом зависят от контекста применения программных решений и особенностей образовательного процесса, в который они интегрированы. Существенная доля таких решений относится к нише дополнительного и корпоративного образования. Как отдельную нишу можно выделить сервисы изучения иностранных языков. В высшем образовании присутствие технологий адаптивного обучения сегодня небольшое, поскольку их применение означает в значительной степени реформу образования. Однако

решения существуют, и компании-производители пытаются наращивать скорость их распространения в цифровых средах высших учебных заведений.

Адаптивное обучение распространяется прежде всего в западных вузах, которые прибегают к услугам таких компаний, как *Knewton*¹ и *Cerego*². Продукты последних позволяют реализовывать адаптивное обучение для учебных дисциплин в поддержку традиционному обучению. При этом популярные западные сервисы для высшего образования пока не вышли на российский рынок по причине недостаточной локализации (или ее отсутствия), а также невозможности технически интегрировать предлагаемые программные решения в инфраструктуру электронной образовательной среды университетов.

Решения, описываемые в настоящей статье, являются проектом Томского государственного университета систем управления и радиоэлектроники (ТУСУР). Цель проекта — предоставить отечественным вузам возможность реализовывать адаптивное электронное обучение и внедрять его в образовательный процесс.

1. Алгоритм адаптивного обучения

1.1. Задача

Основная идея адаптивного обучения заключается в построении оптимальной траектории изучения студентом модулей курса. Модуль — это логически завершенная минимальная единица информации образовательного характера, которая раскрывает одно или несколько терминов или понятий и находится в связи с другими единицами [Кречетов, Кручинин, 2017. С. 75–80]. Модуль может быть представлен текстом, содержать графику, видео или аудио, а также любые иные интерактивные формы представления информации.

Построение траектории из модулей является многокритериальной задачей оптимизации. Однако, учитывая специфику учебных программ вуза, а именно то, что время, отведенное для изучения курса, фиксировано, критерием оптимальности можно считать получение максимального уровня знаний на момент окончания курса при минимальном времени изучения модулей курса, или

$$(1) \quad F(P T_{con}) = \frac{TM}{R(T_{con})} \rightarrow \min.$$

Здесь P — траектория обучения (порядок изучения модулей), TM — суммарное время изучения модулей, R — уровень остаточных знаний. Так как время окончания курса T_{con} является константой, его можно опускать при записи целевой функции (1).

¹ <https://www.knewton.com>

² <https://www.cerego.com>

Задача (1) относится к целочисленному программированию, поскольку последовательность изучения модулей — это последовательность их идентификаторов, представленных целыми значениями. Более того, пространство решений задачи является сильно дискретизированным: как будет показано ниже, далеко не любая последовательность модулей курса является допустимой. Поэтому ограничения задачи можно записать только в терминах дискретной математики как отношения на множествах. Классические оптимизационные алгоритмы плохо подходят для решения таких задач, поэтому для ее решения был выбран генетический алгоритм.

Для экстраполяции уровня остаточных знаний на момент окончания курса, отталкиваясь от промежуточных результатов тестирования, была использована модель, основанная на скорости забывания информации.

Других достаточно обоснованных моделей, пригодных для численного прогнозирования на практике уровня знаний обучающихся в будущие периоды времени на основе их предыдущей истории обучения, практически не существует. Использование байесовских сетей не дает более высокой точности прогнозирования (за пределами задач адаптивного тестирования), зато требует существенных вычислительных ресурсов [Хлопотов, 2014. С. 40–52]. Хорошую скорость принятия решений дает использование технологии машинного обучения (как, например, в *Snappet*³), но для достижения достаточной точности прогнозов в учебном курсе, содержащем хотя бы 150–200 модулей, потребуется база из десятков тысяч уже пройденных траекторий обучения. Поэтому на начальном этапе внедрения учебного курса необходимо использовать статистические модели (например, байесовские сети доверия), или модели, основанные на скорости забывания информации (например, [Харитонов, Крушель, 2012]).

Выкладки используемой в работе математической модели скорости забывания информации, а также модель предметной области приведены в приложении.

Общая блок-схема алгоритма адаптивного обучения дана на рис. 1 [Кречетов и др., 2018. С. 33–40].

Приведем его описание.

1. Формирование выборки. На данном этапе формируется множество A , состоящее из модулей, реализующих недостаточно изученные компетенции (см. формулы 11, 12 в приложе-

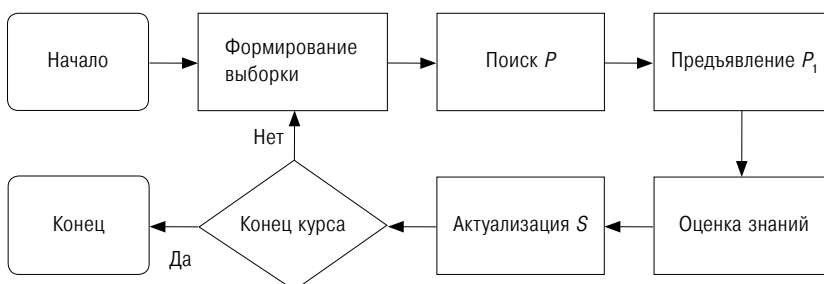
³ <https://nl.snappet.org/>

1.2. Методологические подходы

1.3. Методология, алгоритм

1.3.1. Алгоритм адаптивного обучения

Рис. 1. **Общая блок-схема алгоритма адаптивного обучения**



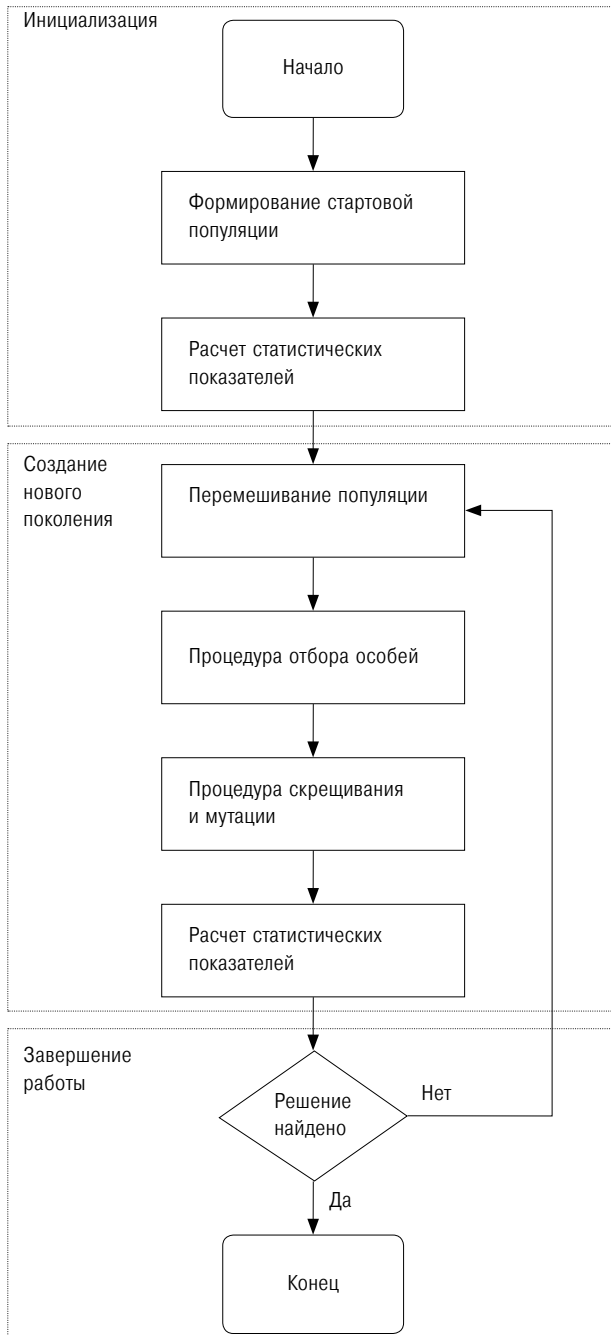
- нии). Компетенция K_j считается недостаточно изученной в двух случаях. Во-первых, если ранее она студентом вообще не изучалась, т. е. $HR_j = \emptyset$. Во-вторых, если компетенция является утраченной, т. е. уровень ее освоения, согласно кривой забывания, со временем опустился ниже уровня R_{norm} .
2. Поиск P . Для поиска траектории обучения используется генетический алгоритм, описанный ниже.
 3. Предъявление P_1 . Студенту предоставляется для обучения первый модуль из P . Учебные модули реализуются в среде дистанционного обучения.
 4. Оценка знаний. В рамках системы дистанционного обучения формируется тест для проверки уровня знаний по выходным компетенциям модуля.
 5. Актуализация S . После проведения теста обновляется актуальный уровень знаний студента в истории HR_j .
 6. Проверка на окончание курса. Окончание курса наступает в двух случаях. Во-первых, при истечении времени курса, т. е. если $t_{тек} \geq t_{кон}$. Во-вторых, если все компетенции изучены на удовлетворительном уровне, т. е. $KS = K$, $KF = \emptyset$ (см. формулы 9, 10 в приложении).

1.3.2. Описание генетического алгоритма

Базовые положения генетического алгоритма генерации последовательности образовательных модулей изложены в [Кречетов, 2014. С. 200–206]. Его общая блок-схема показана на рис. 2.

В классическом генетическом алгоритме популяция описывается как совокупность особей, каждая из которых представляет возможное решение задачи. У каждой особи есть набор хромосом, определяющих ее фенотип. При этом хромосомы обычно представляют собой битовый массив, которым относительно легко можно манипулировать при проведении процедур скрещивания, мутации и т. п. При решении задачи генерации адаптивного контента фенотип особи должен определять траек-

Рис. 2. Общая блок-схема генетического алгоритма



торию обучения P , т. е. порядок изучения модулей из выборки A . Закодировать его в виде битового массива не представляется возможным, так как не любая комбинация битов будет соответствовать допустимому порядку следования модулей. Поэтому в реализованном варианте генетического алгоритма хромосомы описывают упорядоченные цепочки модулей, что повлекло за собой его существенное усложнение.

1.3.3. Исследование и промежуточные результаты

Рассмотрим работу процедур, составляющих генетический алгоритм. Параметрами алгоритма являются:

- размер популяции n_{pop} (выбран равным 100 особям);
- максимальное число поколений n_{gen} (выбрано значение 100);
- вероятность мутации $n_{mutation}$ (1%);
- вероятность скрещивания p_{cross} (90%);
- вероятность появления элитной особи p_{elite} (5%).

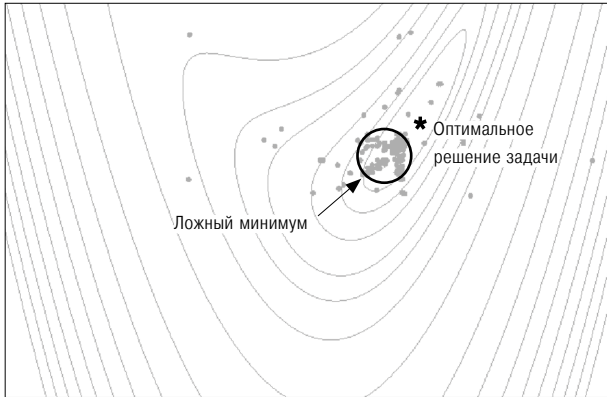
1. Формирование стартовой популяции. Генерируем n_{pop} особей. Каждая особь стартовой популяции генерируется однотипно.
2. Расчет статистических показателей заключается в выполнении двух шагов. Во-первых, здесь нужно для каждой особи оценить функцию пригодности (или целевую функцию, ЦФ). Введенная ранее ЦФ (1) с использованием принятых обозначений запишется как

$$F(P) = \frac{\sum_i TM_{P_i}}{\sum_j R_j} \rightarrow \min,$$

где $i = 1, 2, \dots, np$, $j = 1, 2, \dots, m$. То есть при минимизации ЦФ $F(P)$ мы пытаемся, с одной стороны, найти последовательность модулей P с минимальной общей продолжительностью изучения, с другой — максимизировать суммарный уровень знаний по всем компетенциям K_j на момент окончания курса. Во-вторых, считаем для статистики такие показатели, как минимальное, максимальное и среднее значение ЦФ в рамках популяции (F_{min} , F_{max} , F_{avg}). В качестве приближения оптимального решения выбирается особь с наименьшим значением ЦФ.

3. Перемешивание популяции. В реализованном варианте генетического алгоритма скрещивание происходит между «соседними» особями популяции. Поэтому для обеспечения генетического разнообразия перед скрещиванием особи в популяции перемешиваются случайным образом.
4. Отбор особей. Затем «соседние» особи сравниваются парно, и лучшая из них (имеющая меньшее значение ЦФ)

Рис. 3. Ложная сходимость генетического алгоритма



попадает в новое поколение особей. Таким образом можно получить только половину популяции, поэтому далее производится еще одно перемешивание, и процедура отбора повторяется.

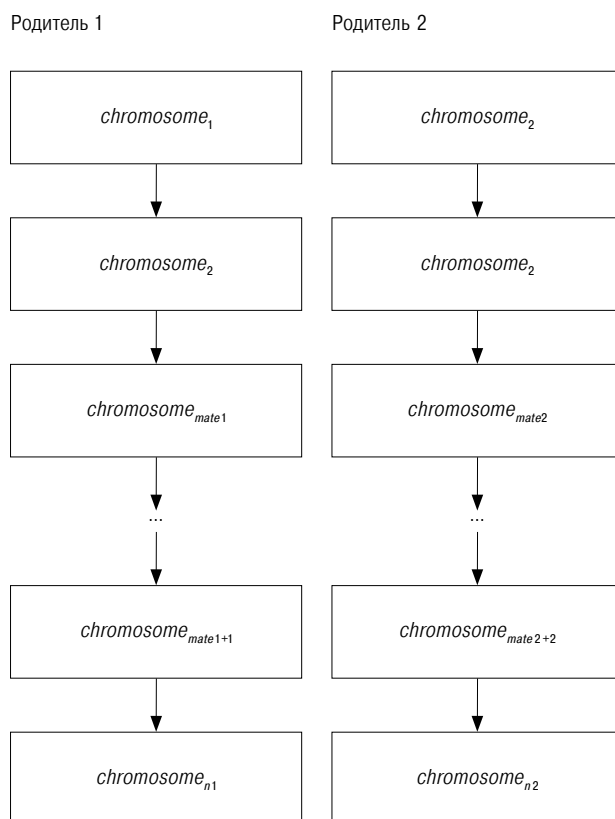
5. Скрещивание и мутация. Процедура скрещивания выполняется в два шага.

- Шаг 1. Выделяем элитные особи. Необходимость их ввода в алгоритм поясняет рис. 3.

Начальное исследование работы генетического алгоритма осуществлялось на решении оптимизационных задач, а именно на минимизации функции Розенброка. Это функция выраженного овражного типа, поэтому многие оптимизационные алгоритмы не могут обеспечить для нее глобальную сходимость. В результате выяснилось, что зачастую генетический алгоритм обеспечивает ложную сходимость. Если вблизи какой-либо точки создавалась высокая концентрация особей популяции, то их генетический материал начинал перевешивать генетический материал остальных особей. То есть даже более удачные особи, расположенные ближе к оптимальной точке, в результате скрещивания с «мусорным» генетическим материалом становились менее пригодными. В итоге через несколько поколений практически вся популяция вырождалась в точку ложного минимума.

Чтобы решить эту проблему, в популяцию были введены элитные особи. Это небольшое число особей популяции (p_{elite}), имеющих максимальное значение функции приспособленности (или минимальное значение ЦФ).

Рис. 4. Иллюстрация скрещивания особей



- Шаг 2. Выбираем две произвольные особи в старой популяции (в предыдущем поколении), и получаем из них две особи новой популяции. Если в скрещивании участвует хотя бы одна элитная особь, то ее хромосомы переходят потомкам следующего поколения без изменений. В противном случае с вероятностью p_{cross} выполняется одноточечное скрещивание.

Для этого сначала выбирается произвольная точка сечения $mate_1$ для первой родительской особи.

Затем несколько раз пробуем найти допустимую точку сечения $mate_2$ для второй родительской особи (рис. 4).

С этой целью выбираем случайным образом точку и проверяем, возможно ли скрещивание двух родительских особей в точках $mate_1$ и $mate_2$ соответственно. Для этого программа формирует «волны» компетенций для каждого родителя — две волны изученных компетенций, распростра-

нящихся от начала цепочек хромосом (KS_1, KS_2), и две волны неизученных компетенций, распространяющихся с конца цепочек (KF_1, KF_2):

$$KS_k = KS \cup \bigcup_{i=1}^{mate_k} KO_i,$$

$$KF_k = \left(KF \cup \bigcup_{i=mate_k+1}^{n_k} KI_i \right) - \left(KS \cup \bigcup_{i=mate_k+1}^{n_k} KO_i \right).$$

Скрещивание возможно, если $KS_1 \subseteq KF_2$ и $KS_2 \subseteq KF_1$.

Если точки скрещивания найти удалось, то выполняем его по схеме на рис. 4.

Эксперименты показывают, что доля удачных скрещиваний находится на уровне 30%.

При выполнении процедуры мутации с вероятностью $p_{mutation}$ происходит одно из двух событий:

- либо замена одного модуля в последовательности другим модулем из базы, если это не нарушает описанных выше условий;
- либо перестановка двух модулей в последовательности при тех же условиях.

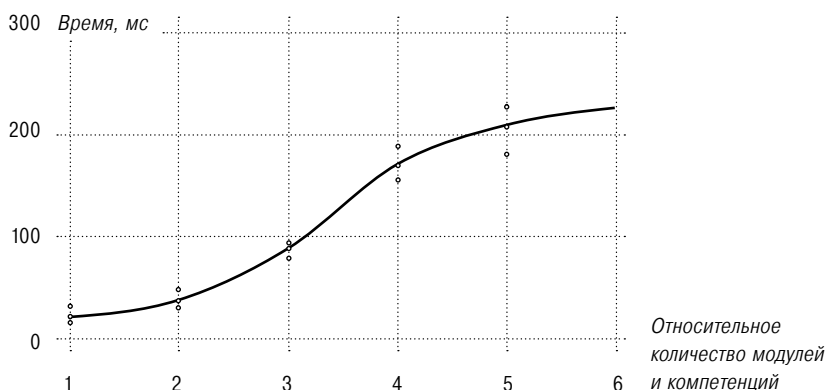
6. Проверка оптимальности решения. Увеличиваем счетчик поколений, и если его значение превышает n_{gen} , то работа алгоритма закончена. Кроме того, считается, что оптимальное решение найдено, если значение F_{min} не изменялось на протяжении последних 10 поколений.

Было выполнено несколько оптимизаций генетического алгоритма, после чего проведено исследование скорости его работы в зависимости от языка реализации и количества модулей и компетенций, входящих в адаптивный курс (рис. 5).

На графике за единицу по оси абсцисс принято условное количество модулей и компетенций в одном из разделов курса «Информатика» (43 модуля, 57 компетенций). Остальные значения были получены при кратном увеличении этого количества. По оси ординат в каждом случае отложено по три точки: минимальное и максимальное время работы алгоритма при данном количестве модулей и компетенций, а также медианное значение, найденное как точка с минимальным среднеквадратичным отклонением:

$$t : \sum_{i=1}^n (t - t_i) \rightarrow \min.$$

Рис. 5. Исследование скорости работы генетического алгоритма



Некоторый разброс в результатах объясняется использованием в работе алгоритма датчика случайных чисел. Поэтому при одних и тех же входных данных может быть сформировано разное количество поколений популяции, разное количество скрещиваний и т. п. Как показывают эксперименты, максимальное количество поколений, которые требуются генетическому алгоритму для нахождения оптимального решения, зависит от вариативности базы модулей и компетенций. Если вариативность не слишком высокая, оптимальное решение может быть найдено уже к 5-му поколению. При повышении вариативности требуемое количество итераций работы алгоритма возрастает.

Таким образом, время работы алгоритма зависит от общего количества модулей и компетенций, имеющихся в курсе, а также от степени вариативности модулей. От общего количества модулей и компетенций в базе данных, а также от общей численности студентов зависимости нет. Проседание производительности сервера генетического алгоритма может наблюдаться, только если к нему поступит одновременно множество запросов от разных клиентов.

Как видно из рис. 5, до некоторой отметки скорость роста функции на графике превышает линейную, но затем замедляется, так как некоторые этапы работы алгоритма, имеющие приблизительно фиксированное время, перестают заметно влиять на общее время его работы. Это дает возможность экстраполировать скорость работы алгоритма для курсов с произвольным количеством модулей и компетенций.

Рис. 6. Часть диаграммы связей (карты памяти) компетенций курса



На этапе планирования практической реализации модели адаптивного обучения были приняты следующие проектные решения.

Во-первых, была выбрана клиент-серверная архитектура. Настольные программные решения в электронном обучении в настоящее время практически не используются. Все вузы, занимающиеся развитием дистанционного обучения, ориентированы именно на онлайн-системы.

Во-вторых, весь инструментарий для создания и запуска адаптивных учебных курсов решено выполнить в форме облачного сервиса типа SaaS (англ. *software as a service* — программное обеспечение как услуга). Это позволит предоставлять данный инструментарий всем заинтересованным вузам и другим организациям, независимо от типа используемой ими системы дистанционного обучения, а также упростит сопровождение и поддержку программной составляющей сервиса.

В-третьих, положено, что модули и компетенции изначально не принадлежат какому-то определенному курсу, т. е. являются универсальными. Все они хранятся в общей базе данных, и при необходимости их можно использовать при формировании того или иного учебного курса.

Практическая реализация заключалась в выполнении следующих задач.

1. Разработана база модулей и компетенций из предметной области дисциплины «Информатика». Описание модулей изначально было выполнено в виде обычного документа *Word*, а компетенций — в виде диаграммы связей (или карты памяти, англ. *mind map*), разработанной в программе *FreeMind* (рис. 6).

2. Практическая реализация

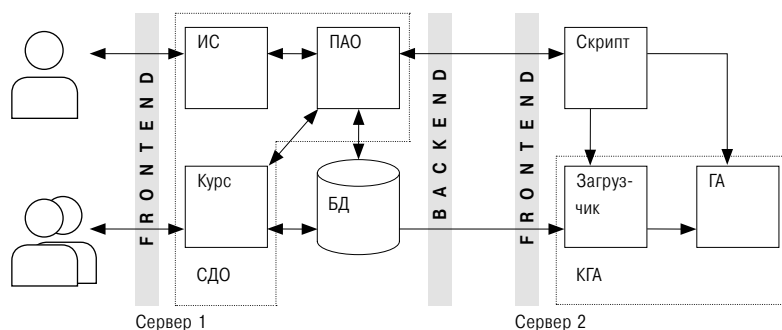
2.1. Результаты согласно целям

2. Выполнена программная реализация описанного ранее генетического алгоритма. Как показали опыты, скриптовые языки типа PHP не подходят для решения данной задачи, поэтому в настоящее время используются реализации на языках C++ и C#.
3. Модифицирована используемая в системах дистанционного обучения база данных, чтобы в ней можно было хранить все описанные в модели предметной области данные.
4. Для системы дистанционного обучения *Moodle* был разработан ряд плагинов для создания и эксплуатации адаптивных курсов. Во-первых, это локальный плагин, который содержит ядро решения, а также интерфейс для заполнения базы. Во-вторых, это плагин типа «Тема», который подгружает ядро и отвечает за автоматическое перемещение по элементам внутри курса для студента. В-третьих, это плагин типа «Ресурс», который отвечает за предоставление контента модулей.
5. Разработанные ранее модули и компетенции с использованием нового инструментария были перенесены в базу данных (рис. 7).
6. Реализация генетического алгоритма была вынесена на отдельный сервер в виде SaaS-решения. В будущем остальные инструментальные средства планируется также перенести в облако.
7. Создан тестовый курс «Информатика», и на нем проведена апробация работы всей системы.

Общая архитектура полученного программного комплекса изображена на рис. 8. Здесь:

- СДО — используемая система дистанционного обучения (*Moodle*);
- Курс — адаптивный учебный курс;
- БД — база данных (не входит в состав *Moodle*, поэтому изображена вне СДО), в которой хранится вся информация о модулях курса, тестах, профилях студентов и т. д.;
- ПАО — плагины для *Moodle*, реализующие модель адаптивного обучения;
- ИС — инструментальная система для разработки адаптивного курса;
- КГА — компонент, реализующий генетический алгоритм;
- Скрипт — скрипт на языке PHP для обмена информацией между серверами;
- Загрузчик — компонент, загружающий из базы данных необходимую для работы генетического алгоритма информацию (профиль пользователя, списки модулей и компетенций и др.);
- ГА — программная реализация генетического алгоритма.

Рис. 8. Общая архитектура программного комплекса



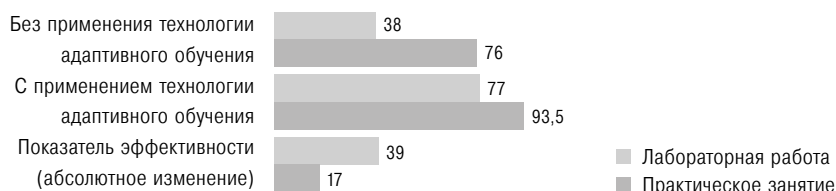
Данная архитектура позволяет гибко менять конфигурацию решения. Она сделана максимально независимой от используемой системы дистанционного обучения — при смене системы необходимо переработать лишь плагин адаптивного обучения и загрузчик генетического алгоритма (если меняется структура базы данных). После переноса всего инструментария в облако все указанные модификации будут выполняться на стороне облачного сервера.

Реализация генетического алгоритма в виде облачного решения удобна также для балансировки нагрузки: работа алгоритма требует существенных вычислительных ресурсов, поэтому использование облачного сервиса позволяет не увеличивать отклик сервера, на котором функционирует система дистанционного обучения. Однако технических препятствий к тому, чтобы расположить систему и весь требуемый инструментарий на одном и том же сервере, если он достаточно мощный, нет.

В 2018 г. между ТУСУР и Национальным исследовательским технологическим университетом «МИСиС» было заключено соглашение, в рамках которого для МИСиС на базе разработанной технологии был создан адаптивный обучающий курс по дисциплине «Общая химия» [Кречетов, Дорофеева, Дегтярев, 2018. С. 76–86]. В осеннем семестре 2018/2019 учебного года кафедрой ОиНХ НИТУ «МИСиС» была проведена апробация курса. При преподавании дисциплины, в отличие от онлайн-курса «Информатика», использовалась модель смешанного обучения, основанная на технологии перевернутого класса. Традиционная структура курса «Общая химия» включает три вида аудиторных занятий: лекции, практические и лабораторные занятия. В ходе реализации курса каждый вид аудиторных занятий дополнился адаптивной составляющей, реализуемой в электронной среде.

2.2. Апробация разработки

Рис. 9. Сравнительная диаграмма успеваемости обучающихся из разных групп



Эффективность адаптивного курса можно оценить на основе диаграммы, представленной на рис. 9. По оси ординат приведена доля обучающихся, успешно справившихся с контрольным мероприятием по виду деятельности (первичная сдача, в процентах). В качестве испытуемых были выбраны группы из одного потока одного направления подготовки, обучающиеся у одного преподавателя.

Из рис. 9 видно, что успеваемость обучающихся из экспериментальной группы (с применением адаптивного обучения) по всем видам контрольных мероприятий значительно выше, чем у студентов, осваивающих программу в традиционном формате. Полное описание результатов апробации приведено в [Кречетов, Дорофеева, Дегтярев, 2018. С. 76–86].

Таким образом, внедрение адаптивного курса в образовательный процесс дало хорошие результаты. Было принято решение о продолжении сотрудничества. В настоящее время ведется разработка для НИТУ «МИСиС» адаптивного курса по дисциплине «Физика», а также разработка адаптивного выравнивающего курса по математике для МИСиС и ТУСУР. Выравнивающий курс по математике необходим студентам 1-го года обучения, которые приходят в вуз с разным уровнем знаний и поэтому не могут одинаково эффективно усваивать вузовский курс высшей математики. Данный курс позволяет подтянуть отстающих студентов до необходимого уровня.

Также происходит доработка модели адаптивного обучения в части сбора данных для составления портрета обучаемого с целью включения больших параметров при построении индивидуальных траекторий обучения. Еще больше возможностей открывают большие данные о том, как студенты работают с контентом, какие траектории являются наиболее эффективными для тех или иных условных категорий студентов, насколько рационально студенты расходуют отведенное на обучение время и т. д.

Предлагаемая технология универсальна и пригодна для разных контекстов применения. С ее помощью можно организовать автономную систему, в которой обучение происходит без преподавателя. С другой стороны, технология может служить эффективным инструментом для составления портрета обучаемого в электронном (заочном) обучении и предоставить преподавателю обширную аналитику о том, как учатся студенты, а также возможность корректировать траектории обучения. А если образовательный процесс протекает с частичной поддержкой аудиторных занятий электронными средствами обучения (в большинстве случаев путем применения систем дистанционного обучения), то результаты очных занятий могут быть внесены в систему для учета последних достижений студентов, и на основе этих данных алгоритмы сформируют актуальную траекторию в системе дистанционного обучения.

3. Заключение

1. Буймов А. Г., Буймов Б. А. (2010) Вероятностная модель эффекта повторений в обучении // Доклады ТУСУР. № 1. С. 236–242.
2. Кречетов И. А. (2014) Алгоритм генерации последовательности образовательных модулей в технологии получения адаптивного образовательного контента // А. В. Колесников (ред.) Гибридные и синергетические интеллектуальные системы. Материалы докладов II Международного Пospelовского симпозиума (Светлогорск, 14–19 мая 2014). С. 200–206.
3. Кречетов И. А., Дорофеева М. Ю., Дегтярев А. В. (2018) Раскрываем потенциал адаптивного обучения: от разработки до внедрения. Материалы Международной конференции «eLearning Stakeholders and Researchers Summit 2018». М: Изд. дом ВШЭ. С. 76–85.
4. Кречетов И. А., Кручинин В. В. (2017) Об одном алгоритме адаптивного обучения на основе кривой забывания // Доклады ТУСУР. № 1. С. 75–80.
5. Кречетов И. А., Романенко В. В., Кручинин В. В., Городович А. В. (2018) Реализация адаптивного обучения: методы и технологии // Открытое и дистанционное образование. № 3. С. 33–40.
6. Ланге В. Н. (1983) О скорости забывания // Вопросы психологии. № 4. С. 142–145.
7. Норенков И. П., Соколов Н. К. (2009) Синтез индивидуальных маршрутов обучения в онтологических обучающих системах // Информационные технологии. № 3. С. 74–77.
8. Норенков И. П., Соколов Н. К., Уваров М. Ю. (2009) Адаптивные среды создания образовательных ресурсов // Наука и образование. Электронный журнал. № 3. <http://technomag.bmstu.ru/doc/115688.html>
9. Норенков И. П., Уваров М. Ю. (2005) База и генератор образовательных ресурсов // Информационные технологии. № 9. С. 60–66.
10. Растринин Л. А., Эренштейн М. Х. (1988) Адаптивное обучение с моделью обучаемого. Рига: Зинатне.
11. Рыбина Г. В. (2014) Системы, основанные на знаниях. Интегрированные экспертные системы. М.: Научтехлитиздат.
12. Рыбина Г. В. (2011) Интеллектуальные обучающие системы на основе интегрированных экспертных систем: опыт разработки и исполь-

Литература

- зования // Информационно-измерительные и управляющие системы. № 10. С. 4–16.
13. Рыбина Г. В. (2010) Современные подходы к реализации интеллектуального компьютерного обучения на основе разработки и использования обучающих интегрированных экспертных систем // Приборы и системы. Управление, контроль, диагностика. № 5. С. 10–15.
 14. Рыбина Г. В. (2008а) Обучающие интегрированные экспертные системы: некоторые итоги и перспективы // Искусственный интеллект и принятие решений. № 1. С. 22–46.
 15. Рыбина Г. В. (2008б) Теория и технология построения интегрированных экспертных систем. М.: Научтехлитиздат.
 16. Харитонов И. М., Крушель Е. Г. (2012) Прогнозирование уровня освоения компетенций выпускниками вуза на основе рейтинговых оценок абитуриентов // Современные проблемы науки и образования. № 6. <http://www.science-education.ru/ru/article/view?id=7531>
 17. Хлопотов М. В. (2014) Модели и алгоритмы интеллектуального анализа образовательных данных для поддержки принятия решений: дис. ... канд. техн. наук. СПб.: Санкт-Петербургский национальный исследовательский университет информационных технологий, механики и оптики.
 18. Brusilovsky P. (2001) Adaptive Hypermedia // User Modeling and User-Adapted Interaction. Vol. 11. No 1–2. P. 87–110.
 19. Brusilovsky P. (1998) Methods and Techniques of Adaptive Hypermedia // Adaptive Hypermedia and Hypermedia. Dordrecht: Kluwer Academic Publishers. P. 1–43.
 20. Brusilovsky P. (1997) Efficient Techniques for Adaptive Hypermedia // C. Nicholas, J. Mayfield (eds) Intelligent Hypertext: Advanced Techniques for the World Wide Web. Lecture Notes in Computer Science. Berlin: Springer-Verlag. Vol. 1326. P. 12–30.
 21. Brusilovsky P. (1996) Methods and Techniques of Adaptive Hypermedia // User Modeling and User-Adapted Interaction. Vol. 6. No 2–3. P. 87–129.
 22. Ebbinghaus H. (1913) Memory: A Contribution to Experimental Psychology. <https://archive.org/stream/memorycontributi00ebbinghaus>

Приложение *Скорость забывания информации.* Первое экспериментальное исследование памяти и процессов забывания выполнил немецкий психолог Герман Эббингауз. Его результаты были опубликованы в 1885 г. в книге «О памяти» [Ebbinghaus, 1913]. Эббингауз показал, что скорость забывания удовлетворительно аппроксимируется функцией

$$(2) \quad R(t) = \frac{k}{\lg t + c},$$

где нужно положить $k = 1,84$ и $c = 1,25$, если время t выражено в минутах [Ланге, 1983. С. 142–145]. Эти значения были получены для запоминания информации, лишенной ассоциативной связи.

Примечание. Уровень знаний не может быть ниже 0 и выше 100%, поэтому должно выполняться условие $R(t) \in [0, 1]$. Выражение (2) при достаточно больших значениях t дает только положительные значения, так как

$$\lim_{t \rightarrow \infty} \frac{k}{\lg t + c} = 0.$$

Однако при малых значениях t выражение плохо обусловлено и может давать произвольный результат от $+\infty$ до $-\infty$. Поэтому на практике для вычисления (2) авторы предлагают использовать следующий подход:

$$R(t) = \begin{cases} 1, & t < 1 \\ \min\left(\frac{k}{\lg t + c}, 1\right), & t \geq 1. \end{cases} \quad (3)$$

Таким образом, $R(t) = 1$ при $\lg t + c \leq k$. Это выражение можно интерпретировать следующим образом: некоторое время после изучения информации она не забывается, т.е. полностью сохраняется в памяти. Это время тем дольше, чем больше коэффициент k .

Имея две экспериментальные точки на кривой забывания, можно определить значения коэффициентов k и c :

$$\begin{cases} R_1 = \frac{k}{\lg t_1 + c}, \\ R_2 = \frac{k}{\lg t_2 + c}, \end{cases}$$

откуда

$$c = \frac{R_2 \lg t_2 - R_1 \lg t_1}{R_1 - R_2}, \quad (4)$$

$$k = R_1(\lg t_1 + c) \text{ или } k = R_2(\lg t_2 + c). \quad (5)$$

Более поздние исследования Эббингауза, проведенные для изучения процесса забывания осмысленной информации, дали следующий процент воспроизведений: спустя один день — 75%, через четыре дня — 70%. Нас больше интересуют последние результаты, так как учебные материалы относятся к осмысленной информации.

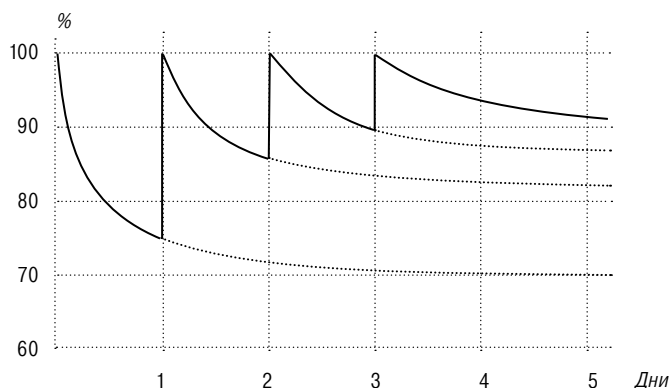
Учитывая, что сутки состоят из 1440 минут, из (4) и (5) получим

$$\frac{0,7 \cdot \lg 5760 - 0,75 \cdot \lg 1440}{0,75 - 0,7} \approx 5,270,$$

$$k = 0,75 \cdot (\lg 1440 + 5,270) \approx 6,322,$$

что можно принять за значения данных коэффициентов по умолчанию, уточняя их по мере тестирования уровня остаточных знаний студента.

Рис. 10. Кривая забывания при итеративном обучении



В данном примере был рассмотрен обобщенный сценарий процесса обучения студента. Очевидно, что приобретение и забывание информации — сложные умственные процессы, результат влияния на которые в настоящее время невозможно спрогнозировать с предельной точностью. Однако в целом неоспоримым является эффект, которого можно достичь за счет итеративного научения, а именно: зафиксировать знания обучаемого к концу обучения на требуемом уровне. Так, в работе [Буймов, Буймов, 2010. С. 236–242] авторы рассматривают разработку вероятностной модели принципа повторений и выводят формулу обобщенной функции забывания, сформированной в результате произвольного числа циклов обучения. Вид этой зависимости изображен на рис. 10.

Таким образом, при повторном изучении информации объем остаточных знаний на момент окончания курса увеличится пропорционально некоторому коэффициенту α :

$$(6) \quad R(t, r) = R(t) \cdot \alpha(r),$$

где r — количество повторений.

Вид зависимости $\alpha(r)$ неизвестен, кроме того, она будет индивидуальной для каждого студента. Предположим, что она имеет вид

$$(2) \quad \alpha(r) = \frac{1,5}{1 + \exp\left(-\frac{r}{1,5}\right)},$$

т. е. в пределе, при достаточно большом количестве повторений, значение $\alpha(r)$ стремится к 1,5 (как уже было сказано, данные до-

пущения будут использованы только для вычисления значений коэффициентов зависимости $R(t, r)$ по умолчанию, а в дальнейшем они будут уточняться на основе фактических результатов обучения студента).

Определим параметры зависимости $R(t, r)$. Предположим, что через минимальное время $t = 1$ мин уровень знаний студента сохраняется на уровне 100%, тогда из (2) и (6) имеем

$$R(1, r) = R(1) \cdot 1 = \frac{k(r)}{\lg 1 + c(r)} = 1 \Rightarrow k(r) = c(r). \quad (7)$$

На момент окончания курса

$$R(t_{\text{кон}}, r) = \frac{k(r)}{\lg t_{\text{кон}} + c(r)} = R(t_{\text{кон}}) \cdot \alpha(r).$$

С учетом (7) получим

$$k(r) = c(r) = \frac{R(t_{\text{кон}}) \cdot \alpha(r)}{1 - R(t_{\text{кон}}) \cdot \alpha(r)} \cdot \lg t_{\text{кон}}.$$

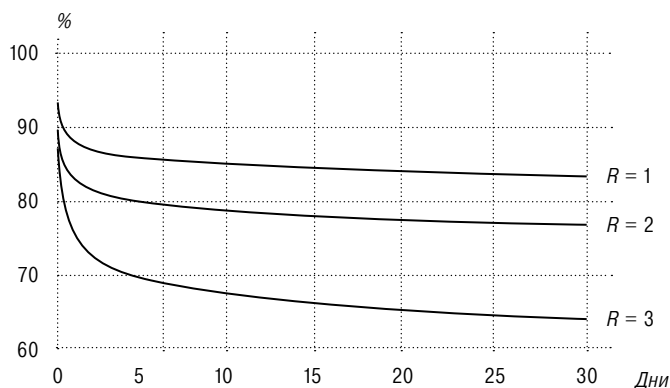
Таким образом, окончательно получим

$$R(t, r) = \begin{cases} R(t), & r = 1, \\ \frac{k(r)}{\lg 1 + c(r)}, & r > 1. \end{cases} \quad (8)$$

При вычислении данного выражения также необходимо использовать подход (3). Вид кривой забывания для разного количества повторений учебного материала r приведен на рис. 11.

Эта формула, однако, была получена исходя из предположения, что при каждом повторении материала студент восстанавливает уровень знаний до 100%. На практике это не так, и последующее тестирование может показать меньший результат. Например, на рис. 12 рассматривается ситуация, когда студент после первой итерации обучения выполнил выходной тест с результатом 100%, а после второй итерации — с результатом 80%. К этому моменту убывание уровня знаний $R(t, 1)$ уже замедлилось, и в дальнейшем кривая убывает уже не так существенно (средняя кривая), но новая кривая $R(t, 2)$ вначале убывает с высокой скоростью, и получается, что в итоге повторное изучение материала привело к тому, что студент стал забывать его быстрее (нижняя кривая). Поэтому в программе формула (8) применяется не ко всему объему полученных знаний, а только к новым полученным знаниям. Так, для ситуации на рис. 12 на момент повторного изучения материала прогнозируемый объем остаточных знаний составлял 75%. Повторное изучение показало, что реально эти знания находятся на уровне 80%, поэтому далее кривая будет вычисляться по формуле

Рис. 11. Поведение кривой забывания при разных значениях параметра r



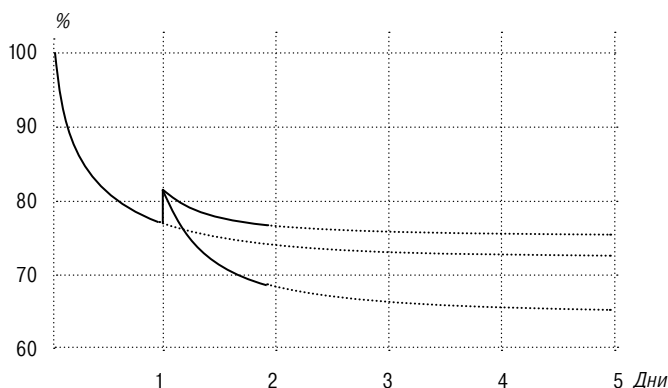
$$R = 100\% \cdot (R(t - t_1, 1) + (80\% - 75\%) \cdot (R(t - t_2, 2)),$$

где t_1 и t_2 — время первого и повторного изучения материала соответственно (верхняя кривая). В реальности формула получается еще сложнее, так как нужно учитывать все итерации изучения материала, индивидуальные корректировки коэффициентов кривой забывания, исходя из полной истории изучения студентом учебного материала, и т. д.

Модель предметной области. Теоретическая модель предметной области описана в [Кречетов, Кручинин, 2017. С. 75–80]. В процессе практической реализации она подверглась некоторым изменениям [Кречетов и др., 2018. С. 33–40]. Введем следующие обозначения для дальнейшего описания алгоритма адаптивного обучения.

1. Множество компетенций курса $K = K_j$, $j = 1, 2, \dots, m$, где m — количество компетенций (в работе [Кречетов, Кручинин, 2017. С. 75–80] было введено понятие субкомпетенций, однако здесь для краткости будем называть их просто компетенциями), а также множество важных компетенций $IK \subset K$ (которые должны быть освоены в обязательном порядке).
2. Множество образовательных модулей курса $M = M_i$, $i = 1, 2, \dots, n$, где n — количество модулей. Модуль описывается четверкой $M_i = (TM_i, KI_i, KO_i, RO_i)$, $i = 1, 2, \dots, n$, где TM_i — время изучения модуля, KI_i и KO_i — списки входных ($KI_i = \{comp_k\}$, $comp_k \in [1, m]$, $k = 1, 2, \dots, ni$) и выходных ($KO_i = \{comp_k\}$, $comp_k \in [1, m]$, $k = 1, 2, \dots, no$) компетенций,

Рис. 12. Кривая забывания при итеративном обучении



RO_i — получаемый уровень знаний для каждой выходной компетенции.

3. Множество групп тестов $T = \{T_j\} = \{\{T_{jk}\}\}$, $j = 1, 2, \dots, m$, $k = 1, 2, \dots, nt_j$, предназначенных для оценки знаний компетенций K . При этом группа тестов T_j предназначена для проверки знаний по компетенции K_j и может включать один или множество тестов T_{jk} .
4. Время начала курса $t_{нач}$, время окончания курса $t_{кон}$, текущее время $t_{тек}$.
5. Профиль студента $S = (HM, HT, HR, RK)$. Здесь компонент HM — это личная история изучения студентом модулей, HT — история тестирования студента, HR — история изменения уровня знаний по каждой компетенции, RK — это коэффициенты кривой забывания каждой компетенции для разного количества повторений студентом учебного материала.
6. Уровень удовлетворительного усвоения учебного материала R_{norm} .
7. Множества достаточно (KS) и недостаточно (KF) изученных компетенций. Если уровень RA_{jl} усвоения знаний по какой-либо компетенции K_j ниже R_{norm} , то она считается недостаточно изученной, в противном случае — достаточно изученной. То есть в терминах исчисления предикатов

$$KS = \{comp_k \mid RA_{comp_k l} \geq R_{norm}\}, comp_k \in [1, m], k = 1, 2, \dots, ns, \quad (9)$$

$$KF = \{comp_k \mid RA_{comp_k l} < R_{norm}\}, comp_k \in [1, m], k = 1, 2, \dots, nf. \quad (10)$$

8. Множество модулей $A = \{mod_k\}$, $mod_k \in [1, n]$, $k = 1, 2, \dots, na$, реализующих недостаточно изученные компетенции, т. е.

$$(11) \quad \bigcup_{i \in A} KO_i \in KF,$$

но при этом

$$(11) \quad KO_{mod_k} \notin KS, \quad k = 1, 2, \dots, na.$$

Последнее условие означает, что множество A не включает модули, все выходные компетенции которых уже изучены на достаточном уровне.

9. Путь P , описывающий непротиворечивый порядок изучения модулей множества A , $P = \{mod_k\}$, $mod_k \in [1, n]$, $k = 1, 2, \dots, np$, (не обязательно всех). В результате изучения модулей, входящих в P , должен быть повышен уровень знаний по всем компетенциям, для которых $RA_{ji} < R_{norm}$. Требование непротиворечивости означает, что к моменту изучения каждого модуля из P должны быть изучены все требуемые входные компетенции, т. е.

$$KS \cup \bigcup_{i < k} KO_{mod_i} \subseteq KI_{mod_k}, \quad k = 1, 2, \dots, np.$$

Implementing the Adaptive Learning Techniques

Ivan Krechetov

Head of the Laboratory of Instrumental Modelling and Learning Systems, Tomsk State University of Control Systems and Radioelectronics. E-mail: kia@2i.tusur.ru

Authors

Vladimir Romanenko

Candidate of Sciences in Technology, Associate Professor, Department of Automated Control Systems, Tomsk State University of Control Systems and Radioelectronics. E-mail: rva@2i.tusur.ru

Address: Room 607, 146 Krasnoarmeyskaya Str., 634034 Tomsk, Russian Federation.

The concept of adaptive learning emerged a few decades ago, but most theoretical findings have never been put into practice, and software solutions had no significant reach for a long time due to insufficient e-learning technology development and coverage. The recent advancements of information technology allow the elaboration of complex big data analytics and artificial intelligence solutions, in adaptive learning in particular.

Abstract

This article investigates exploitation of adaptive learning technology and techniques. The solutions proposed allow mapping optimal individualized learning paths for students in online courses, using the ratio of the level of knowledge at course completion to time spent on the course as an optimality criterion. A genetic algorithm is used to solve this optimization problem. A model based on the speed of forgetting was applied to extrapolate the level of retained knowledge.

Practical implementation of the technology proposed involves a set of tools to expand the adaptive learning opportunities of distance learning systems and a module to operate the genetic algorithm. We developed a few options of software architecture using different technologies and programming languages and either one or two servers. The solution was tested during the design of adaptive learning courses for National University of Science and Technology MISIS (NUST MISIS) and Tomsk State University of Control Systems and Radioelectronics (TUSUR).

adaptive learning, e-learning, genetic algorithm, distance learning system.

Keywords

Brusilovsky P. (2001) Adaptive Hypermedia. *User Modeling and User-Adapted Interaction*, vol. 11, no 1–2, pp. 87–110.

References

Brusilovsky P. (1998) Methods and Techniques of Adaptive Hypermedia. *Adaptive Hypermedia and Hypermedia*, Dordrecht: Kluwer Academic Publishers, pp. 1–43.

Brusilovsky P. (1997) Efficient Techniques for Adaptive Hypermedia. *Intelligent Hypertext: Advanced Techniques for the World Wide Web. Lecture Notes in Computer Science* (eds C. Nicholas, J. Mayfield), Berlin: Springer-Verlag, vol. 1326, pp. 12–30.

Brusilovsky P. (1996) Methods and Techniques of Adaptive Hypermedia. *User Modeling and User-Adapted Interaction*, vol. 6, no 2–3, pp. 87–129.

Buimov A., Buimov B. (2010) Veroyatnostnaya model efekta povtoreniy v obuchenii [Probabilistic Model of Recurrences]. *Proceedings of Tomsk State University of Control Systems and Radioelectronics*, no 1, pp. 236–242.

- Ebbinghaus H. (1913) *Memory: A Contribution to Experimental Psychology*. Available at: <https://archive.org/stream/memorycontributi00ebbiuoft> (accessed 30 April 2020).
- Kharitonov I., Krushel E. (2012) Prognozirovaniye urovnya osvoeniya kompetentnykh vypusknikami vuza na osnove rejtingovykh otsenok abiturientov [The Prediction of the Graduates Competency Mastering Level Based on the Current Rating Estimates]. *Modern Problems of Science and Education*, no 6. Available at: <http://www.scienceeducation.ru/ru/article/view?id=7531> (accessed 30 April 2020).
- Khlopotov M. (2014) *Modeli i algoritmy intellektualnogo analiza obrazovatelnykh dannykh dlya podderzhki prinyatiya resheniy* [Models and Algorithms of Educational Data Intelligence Analysis to Support Decision-Making Processes] (PhD Thesis). St. Petersburg: ITMO University.
- Krechetov I. (2014) Algoritmy generatsii posledovatelnosti obrazovatelnykh moduley v tekhnologii polucheniya adaptivnogo obrazovatel'nogo kontenta [Learning Module Sequence Generation Algorithms in Adaptive Learning Technology]. *Gibridnye i sinergeticheskie intellektualnye sistemy* [Hybrid and Synergetic Intelligent Systems]. *Proceedings of the 2nd Pospelov International Symposium (Svetlogorsk, May 14–19, 2014)* (ed. A. Kolesnikov), pp. 200–206.
- Krechetov I., Dorofeeva M., Dyagterev A. (2018) Raskryvaem potentsial adaptivnogo obucheniya: ot razrabotki do vnedreniya [Unlocking the Potential of Adaptive Learning: From Design to Implementation]. *Proceedings of the eLearning Stakeholders and Researchers Summit 2018*, Moscow: HSE, pp. 76–85.
- Krechetov I., Kruchinin V. (2017) Ob odnom algoritme adaptivnogo obucheniya na osnove krivoy zabyvaniya [About One Algorithm of Adaptive Learning Based on Forgetting Curve]. *Proceedings of Tomsk State University of Control Systems and Radioelectronics*, no 1, pp. 75–80.
- Krechetov I., Romanenko V., Kruchinin V., Gorodovich A. (2018) Realizatsiya adaptivnogo obucheniya: metody i tekhnologii [Implementation of Adaptive Learning: Methods and Technologies]. *Open and Distance Education*, no 3, pp. 33–40.
- Lange V. (1983) O skorosti zabyvaniya [On the Speed of Forgetting]. *Voprosy Psichologii*, no 4, pp. 142–145.
- Norenkov I., Sokolov N. (2009) Sintez individualnykh marshrutov obucheniya v ontologicheskikh obuchayushchikh sistemakh [Creation of Individual Learning Routes in Ontology Education Systems]. *Information Technology*, no 3, pp. 74–77.
- Norenkov I., Sokolov N., Uvarov M. (2009) Adaptivnye sredy sozdaniya obrazovatelnykh resursov [Adaptive Learning Environments]. *Nauka i obrazovanie / Science & Education*, no 3. Available at: <http://technomag.bmstu.ru/doc/115688.html> (accessed 30 April 2020).
- Norenkov I., Uvarov M. (2005) Baza i generator obrazovatelnykh resursov [The Base and Generator of Learning Content]. *Informacionnye tekhnologii / Information Technologies*, no 9, pp. 60–66.
- Rastrigin L., Erenshteyn M. (1988) *Adaptivnoe obuchenie s modelyu obuchayemogo* [Adaptive Learning with the Learner Model]. Riga: Zinatne.
- Rybina G. (2014) *Sistemy, osnovannye na znaniyakh. Integrirovannye ekspertnyye sistemy* [Knowledge-Based Systems. Integrated Expert Systems]. Moscow: Nauchtekhlitizdat.
- Rybina G. (2011) Intellektualnye obuchayushchie sistemy na osnove integrirovannykh ekspertnykh sistem: opyt razrabotki i ispolzovaniya [Integrated Knowledge-Based Expert Systems: Development and Implementation

- Practices]. *Informatsionno-izmeritelnye i upravlyayushchie sistemy/Information-Measuring and Control Systems*, no 10, pp. 4–16.
- Rybina G. (2010) Sovremennye podkhody k realizatsii intellektualnogo kompyuternogo obucheniya na osnove razrabotki i ispolzovaniya obuchayushchikh integrirovannykh ekspertnykh sistem [Modern Approaches to Intelligent Computer-Aided Instruction Based on the Design and Implementation of Integrated Knowledge-Based Expert Systems]. *Pribory i sistemy. Upravlenie, kontrol, diagnostika/Instruments and Systems: Monitoring, Control, and Diagnostics*, no 5, pp. 10–15.
- Rybina G. (2008a) Obuchayushchie integrirovannye ekspertnye sistemy: nekotorye itogi i perspektivy [Integrated Knowledge-Based Expert Systems: Some Conclusions and Outlooks]. *Iskusstvenny intellekt i prinyatie resheniy/Artificial Intelligence and Decision Making*, no 1, pp. 22–46.
- Rybina G. (2008b) *Teoriya i tekhnologiya postroeniya integrirovannykh ekspertnykh system* [Theory and Technology of Integrated Expert Systems Construction]. Moscow: Nauchtekhlitizdat.