

СИСТЕМЫ ИСКУССТВЕННОГО ИНТЕЛЛЕКТА

И.А. Белоус, Н.В. Белоус, Т.В. Горбач, И.Ю. Шубин⁴

Украина, Харьковский национальный университет радиоэлектроники
shubin@kture.kharkov.ua

РЕАЛИЗАЦИЯ МЕТОДОВ АДАПТАЦИИ В ГИПЕРМЕДИЙНЫХ СИСТЕМАХ ОБУЧЕНИЯ

Введение

Интернет получил широкое распространение в области электронного дистанционного обучения. В данной области разработчики систем дистанционного обучения хотят направлять пользователей через гиперпространство согласно их предпочтениям и уровню знаний. Достичь этой цели можно с помощью технологий адаптивной гипермедиа, которые и рассматриваются в статье.

На данный момент разработано большое количество дистанционных обучающих систем. Системы управления обучением в своем традиционном виде не в состоянии поддерживать все новые возможности Интернет-технологий и использовать потенциал всех средств, взаимодействия внутри сетей. В этой области создаются все более динамичные платформы, которые приходят на смену традиционным. При активном электронном обучении используется широкий спектр Интернет-технологий, что позволяет внедрять недоступные для традиционных видов обучения методики. Эта проблема частично решается разработкой и внедрением адаптивных обучающих систем. Но в реальности создано не так уж много систем подобного класса, которые дей-

⁴ © Белоус И.А., Белоус Н.В., Горбач Т.В., Шубин И.Ю., 2010

ствительно отвечают требованиям адаптивности систем [1, 2].

Адаптивность играет большую роль при разработке систем дистанционного обучения, потому что в процессе дистанционного обучения участвует много различных студентов, каждый из которых имеет свои интересы и знания. Однопользовательские обучающие приложения разрабатываются только для определенного класса пользователей с одним складом ума, что может не подойти другим. Под адаптивностью имеется в виду различные пути прохождения контента разными обучаемыми в обучающей системе.

Адаптивная гипермедиа – технология создания гипертекстовых и гипермедийных систем, которая отражает определенные характеристики пользователя в его модели и применяет эту модель для адаптации различных аспектов системы к потребностям пользователя.

Документ гипермедиа - основной компонент WWW, который позволяет пользователям свободно перемещаться (просматривать) информацию в гиперпространстве и состоит из узлов, содержащих информацию и связи, соединяющие их. Гипермедиа – гипертекст, в котором могут быть ссылки на другой тип информации (аудио, видео, анимацию и т.д.) [2].

Системы адаптивной гипермедиа применяют различные виды моделей пользователя для адаптации контента автоматизированных обучающих систем (АОС) и внутренних ссылок под уровень знаний и интересы пользователя. Образование всегда было одной из главных областей применения адаптивной гипермедиа. Большинство адаптивных систем гипермедиа используют методы, которые позволяют разработчикам описывать навигационные правила перемещения обучаемых по контенту АОС [1, 2].

Поэтому целью данной статьи является разработка упрощенного формата навигационных правил для их применения в АОС при дистанционной форме и инструментального средства для проведения экспериментов с ними. Данная задача является актуальной, поскольку существующие АОС, основанные на применении навигационных правил, не предоставляют разработчикам ни средств, ни функций, которые могут уменьшить сроки и трудоемкость составления навигационных правил.

Решение задачи

При проведении данного исследования использован метод сокрытия связей, потому что разрабатываемая АОС должна иметь возможность направить обучаемого по оптимальному пути обучения (рис. 1).

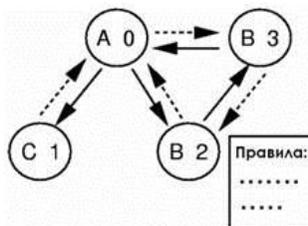


Рис. 1. Гипермедиа модель

Модели обучаемых АОС можно разделить на три вида: оверлейные, стереотипные и модели, которые используют ключевые слова. Оверлейная модель основана на структурной модели предметной области (контента АОС), которая представлена как семантическая сеть понятий. Стереотипная модель назначает пользователю один из нескольких возможных стереотипов для определения соответствующей ему стратегии обучения. Модель обучаемого, использующая ключевые слова, представляет собой вектор или матрицу, элементы которой характеризуют степень интереса в ключевом слове или теме. Системы адаптивной гипермедиа, построенные на основе данной модели позволяют хранить историю взаимодействия обучаемого с системой и относительно просто могут быть преобразованы в строки запросов при реализации модели адаптации [3, 4].

Для разработки АОС предлагаемой архитектуры используем комбинированную модель обучаемого, хранящую как долгосрочную, так и краткосрочную информацию о нем. Для того, чтобы модель обучаемого была более простой, упростим формат навигационного правила. В качестве параметра обучаемого предлагается использовать интересы и уровень знаний пользователя. Это позволит проводить моделирование обучаемого в течение продолжительного времени.

Таким образом, предлагаемая модель обучаемого состоит из последовательности классов узлов. В каждом узле находится содержание, которое показывается обучаемым. Обучаемый может двигаться между узлами по связям между ними. В каждом узле обучаемый имеет возможность выбора связи возврата к предыдущему узлу. Каждый узел идентифицируется номером и классом узла. У всех узлов есть навигационные правила.

Каждому узлу сопоставляем символ, характеризующий его класс. Понятие класса необходимо для формализации навигационных правил и логического вывода на них. Класс также является одной из компо-

нент модели обучаемого. Класс узла определяют разработчики АОС, используя следующие критерии:

- показывает ли АОС фрагмент контента, предназначенный только для определенного вида пользователя?
- предлагает ли узел пользователям объяснение, вопрос, тестовое задание или делает что-то еще в дидактическом плане?
- к каким из категорий контента АОС принадлежит узел (в случаях, когда информация узла может принадлежать более чем к одной категории)?

Предлагаемая структура модели обучаемого содержит два компонента: набор параметров обучаемого и историю взаимодействия с АОС.

Параметры модели обучаемого могут принимать различные значения и характеризовать различные свойства субъекта учебного процесса, такие например как уровень знаний обучаемого. Последний может представлять собой число из некоторого наперед заданного диапазона. Разработчик АОС присваивает значение данному набору параметров, которое используется в навигационных правилах. Необходимо отметить, что указанный набор параметров может быть также задан тьютором или автоматически самой АОС. Например, он может быть установлен по результатам ответа обучаемого на анкетный опрос (тест) или при использовании среднестатистических результатов прохождения тестов несколькими группами обучаемых и т.д.

История взаимодействия обучаемого с АОС представляется как последовательность классов узлов, которые он посетил и информацию с которых он просмотрел.

Предлагаемый навигационный метод основан на адаптивной технологии сокрытия связей (см. выше). АОС, используя навигационное правило, решает, какие узлы скрывать и какие узлы могут быть связаны с текущим.

Представляется целесообразным реализовать следующие четыре вида навигационных правил:

- навигационные правила узла;
- общие навигационные правила;
- локальные пользовательские навигационные правила;
- глобальные пользовательские навигационные правила.

Навигационное правило, которое использует набор параметров обучаемого из его модели, является пользовательским навигационным правилом. Навигационное правило может также быть разделено на два типа: правило узла и общее правило. Правило узла определено и применяется только для определенного узла. Общее правило – для того

чтобы описать наиболее часто встречающиеся навигационные пути в гиперпространстве и часто используемые сегментации диапазона параметров обучаемого.

В навигационном правиле разработчик АОС описывает связи, которые должны быть показаны в соответствии с идентификатором узла или в соответствии с классом узла, который является целью связи. Система скрывает все связи, на которые не ссылаются в навигационном правиле.

Форматы четырех видов навигационного правила следующие:

Навигационное правило узла: $C_n \dots C_i h + \dots + C_m i \dots + C_m h = D_j, \dots, D_n$.

Общее навигационное правило: $C_n \dots C_i h + \dots + C_m i \dots + C_m h = C_i, \dots, C_n$.

Локальное пользовательское навигационное правило:

$$e_j \# P; \# e_2: D_b, \dots, D_n.$$

Глобальное пользовательское навигационное правило:

$$e_i \# P_j \# e_2: C_b^f, \dots, C_n^f,$$

где:

C – класс узла;

D – идентификатор узла, информация которого будет показана обучаемому;

h – количество историй, которые будут задействованы;

m – количество образов пути;

n – количество идентификаторов или классов узлов, которые будут показаны обучаемому;

p – параметр обучаемого;

e – предел параметра обучаемого;

$\#$ – операция, которая представляется одним из следующих трех логических операторов: '<', '<=' или '='.

С левой стороны от символа '=' в первом и втором правилах специфицирован образец истории пути, который представляет собой результат работы обучаемого с контентом АОС. Навигационное правило означает, что система показывает обучаемому связи, идентификатор узла или класс которых указан в его правой части, в том случае, если история пути обучаемого соответствует одному из образцов историй, которые записаны в его левой части. Отличия третьего и четвертого навигационных правил от первого и второго заключаются в том, что слева от символа ':' указан диапазон значений параметра обучаемого. Пользовательское навигационное правило означает, что система показывает связи, идентификатор узла или класс которых указан в его правой части, если параметр обучаемого, указанный в левой части, находится в пределах заданного диапазона.

Если у узла есть несколько навигационных правил, система показывает все связи, которые подтверждаются любым навигационным правилом. Таким образом, если, по крайней мере, одно правило из нескольких правил одобряет показ определенной связи, система показывает связь независимо от других навигационных правил.

Пример навигации. На рисунке 2 показан пример навигации на основе всех описанных выше навигационных правил. Классы определены следующим образом:

A – узлы с вопросом (тестовым заданием);

B – узлы, содержащие информацию для правильного ответа обучаемого;

C – узлы с информацией для неправильного ответа обучаемого;

D – узлы с объяснением для студентов с высоким уровнем знаний;

E – узлы с объяснением для студентов с низким уровнем знаний.

В качестве параметра обучаемого *p1* выступает его уровень знаний по данной дисциплине.

Навигационное правило узла определено для узла №5. Данное правило применимо только в этом узле. "АСА = 7" в этом правиле означает, что, когда пользователь заходит в узел № 5 и история пути пользователя – "АСА", система показывает связь к узлу №7 и скрывает связь к узлу № 8. Поскольку класс В означает, что обучаемый ответил правильно, а класс С – неправильно, история взаимодействия обучаемого с АОС показывает, что пользователь отвечал на вопрос в узле № 0 неправильно, а в узле № 4 – правильно. "АВА=8" означает что, если пользователь ответил правильно на вопросы в узлах № 0 и 4, то система показывает только связь к узлу № 8. То есть система изменяет ход учебного процесса согласно текущим результатам учебной деятельности обучаемого.

В данном примере также задано глобальное пользовательское навигационное правило. Это правило может быть применено в любом узле гиперпространства. В примере на рисунке 2 оно используется в узле № 6. Так, если параметр обучаемого *pi* больше или равен 0, но не превышает 80, система показывает связи к узлам с классом *D* и игнорирует все другие связи. Если параметр *pi* больше 79, но не превышает 100, система показывает связи только к узлам класса *E*. Таким образом, достигается адаптация учебного процесса в АОС к уровню знаний обучаемого.

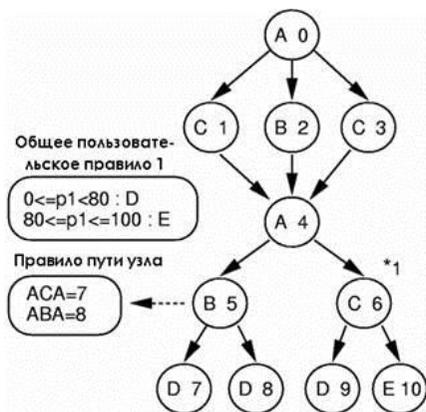


Рис. 2. Примеры навигационных правил

Инструментальное средство исследования навигационных правил предназначено для автоматизации составления навигационных правил разработчиком АОС. Оно является составной частью АОС и помогает разработчику в описании навигационных правил, а также исследует результаты выполнения навигационного правила, прежде чем включить его в узлы. Таким образом, целью создания данного инструментального средства является реализация правильной навигации по контенту АОС с наименьшим количеством ошибок и упрощение составления навигационных правил. Наиболее распространенными навигационными ошибками, которые могут встречаться в контенте любой АОС, являются «тупик» и «петля».

Тупик. Состояние АОС, когда все связи скрыты и обучаемый не может пойти куда-нибудь далее после достижения узла с навигационным правилом. Проблема тупика может быть вызвана плохим навигационным правилом. Появление тупика вынуждает обучаемого прекратить движения по достижению следующих узлов в гиперпространстве. Другими словами, возникает препятствие продвижению обучаемого.

Петля – это состояние АОС, когда обучаемый достигает узла, в котором он уже был. Данная ситуация еще называется «перекручиванием поиска». Использование петли бывает эффективно, например, когда связь необходима для возврата к главной странице. Проблемы с петлями возникают лишь в том случае, когда они являются незапланированными разработчиками АОС. Действительно, поскольку система скрывает динамически связи, которые могли бы вызвать проблемы для навигации обучаемого, это может привести к появлению петель.

Обнаружение тупика. Тупик может быть обусловлен навигационным правилом узла, пользовательским навигационным правилом или другими правилами. Предлагаемый алгоритм обнаружения тупика проверяет возможность появления тупика в узле из-за правил пути или пользовательских правил. Если у узла есть несколько видов навигационных правил, система показывает все связи, которые правило пытается применить. Если инструмент обнаруживает тупик, вызванный одним из видов навигационного правила (например, правило узла) в узле, другой вид правила (например, пользовательское правило) может попытаться показать эти связи в узле. Поэтому, когда алгоритм обнаруживает очевидный тупик в узле, он проверяет, определен ли другой вид правила для него. Если никакое другое правило не определено для узла, значит, обнаружен тупик. Если другой вид правила определен в узле, то существует возможность появления тупика.

Тупик, созданный навигационными правилами, может образоваться в двух случаях: когда образец истории пути, по которому следовал обучаемый, не включен в навигационные правила; когда, ни одна из связей текущего узла не описана в навигационных правилах. Предлагаемый алгоритм не только обнаруживает тупики (возможность появления тупиков), но и определяет образец истории пути, который вызывает тупик.

Алгоритм обнаружения тупиков, основанный на навигационных правилах:

1. Спецификация узла – разработчик АОС определяет узел, который необходимо проверить.

2. Экспертиза визуализируемых связей - система проверяет, существуют ли связи, которые будут показаны согласно образцу истории пути, описанному в навигационном правиле, и определенные в указанном узле гиперпространства. После чего сравниваются узлы, описанные как визуализируемые в навигационных правилах узлов, которые являются непосредственно связанными с текущим.

3. Регистрация так называемого «живого пути» (схемы навигации без тупика) – система распознает образец истории пути, у которого есть связи, которые должны быть показаны. Если обучаемый следует по живому пути к указанному узлу, то гарантировано будут существовать связи для продолжения пути. Каждый живой путь заносится в специальный список, где он ранжируется согласно длине образца истории пути.

4. Система выполняет поиск в глубину, начиная с текущего узла и используя обратные связи, рассматривая таким образом текущий узел как корень перевернутого дерева.

5. Исследование пути – ищем живой путь среди путей, отобранных на предыдущем шаге. Каждый путь, найденный посредством поиска в глубину, проверяется по списку живых путей. Если найден живой путь, возвращаемся на четвертый шаг для продолжения поиска в глубину от верхнего узла. Если найденный путь не является живым, то переходим к шестому шагу.

Обнаружение возможности тупика – если длина текущей глубины поиска достигла максимальной длины путей, зарегистрированных в списке живых путей, то существует возможность появления тупика, когда обучаемый следует по этому пути, и поиск продолжается с седьмого шага. Если максимальная длина не достигнута, возвращаемся на четвертый шаг.

Решение относительно тупика: система проверяет, определено ли правило в любом из узлов на пути. Если никакое навигационное правило не определено ни для одного из этих узлов, то система решает, что тупик появится, когда пользователь будет следовать по этому пути. Если навигационные правила определены, хотя бы для одного узла, то система решает, что есть возможность появления тупика по следованию этого пути пользователем. После этого система возвращается на четвертый шаг.

Тупик, вызванный пользовательским правилом, возникает, когда значение параметров обучаемого выходит за пределы диапазонов, описанных в локальных или глобальных навигационных правилах. Все визуализируемые узлы, описанные в пользовательских правилах, не имеют связи с текущим узлом. Предлагаемый алгоритм не только обнаруживает такие тупики, но также и определяет правило, которое вызывает данный тупик.

Алгоритм обнаружения тупиков, вызванных пользовательскими правилами:

Спецификация узла – разработчик АОС определяет узел, который необходимо проверить.

Исследование визуализируемой связи – система проверяет, существует ли визуализируемая связь для определенного диапазона параметра обучаемого, как описано в пользовательском правиле указанного узла. Данная проверка осуществляется посредством сравнения визуализируемых узлов в пользовательских правилах с узлами, которые непосредственно связаны с текущим. Если такие связи существуют, то данный диапазон считается «живым» (диапазоном, у которого есть визуализируемые связи)

Исследование диапазона параметра – система проверяет, являются ли все диапазоны параметра обучаемого живыми. Если есть диапазон,

который не является живым, следовательно, он и является возможной причиной появления тупика.

Даже если путь определяет петлю, не учитывая технологии сокрытия связей, в действительности она может не являться таковой. Поэтому в определении петель важную роль играет учет технологии сокрытия связей. Поэтому после определения параметров обучаемого и соответствующих навигационных правил, необходимо проверить, остается ли петля петель после сокрытия связей. Описанный ниже алгоритм позволяет обнаружить петли, осуществляя поиск в глубину от определенного узла контента АОС. Предлагаемый алгоритм не только обнаруживает петли, но также и получает путь перемещения обучаемого.

Алгоритм обнаружения петли:

Спецификация узла и максимальной длины – разработчик АОС определяет узел, от которого начинается поиск в глубину на заданную максимальную длину.

Выполнение навигационных правил – система выполняет навигационные правила текущего узла и скрывает связи. После этого регистрирует показанные связи в списке визуализированных связей, который необходим для выполнения поиска в глубину.

Поиск в глубину - осуществляется один шаг поиска в глубину с начала использованных связей, зарегистрированных в списке, полученном на шаге 2.

Исследование петли – система ищет идентификатор узла, который достигнут при поиске в глубину. Если узел с таким идентификатором уже присутствует в истории пути, то данный путь является петлей.

Определение длины – если текущая глубина поиска меньше значения, указанного в шаге 1, то система возвращается к предыдущему узлу и переходим к шагу 3. Иначе возвращаемся к шагу 2.

Во втором шаге предложенного алгоритма, если навигационное правило определено в узле, где система достигла порогового значения глубины поиска, то она не может выполнить это навигационное правило. В таком случае, система не выполняет навигационное правило, а показывает все связи для обнаружения всевозможных потенциальных петель.

Заключение

В данной статье предложен подход к разработке автоматизированных обучающих систем на основе методов адаптивной гипермедиа. Предложенный подход снижает трудоемкость разработки обучающих систем за счет автоматического управления ходом учебного процесса

посредством навигационных правил. Кроме того, в данной статье предложены алгоритмы, реализация которых приводит к уменьшению количества ошибок в описании навигационных правил, для которых был разработан специальный упрощенный формат, что делает их более простыми для восприятия. Созданная на основе предложенного подхода автоматизированная обучающая система управляет маршрутом движения обучаемого по своему контенту с помощью технологии сокрытия связей для оптимизации учебного процесса. Более того, в данной статье разработано инструментальное средство проверки наличия ошибок в созданных навигационных правилах с целью дальнейшего сокращения их количества. Предложенные подходы и алгоритмы могут быть использованы в любых системах адаптивной гипермедиа, которые управляют перемещением пользователя по гиперпространству.

Библиографический список

1. *Bilous N.; Shubin I.; Vyrodov O.* The conception of interactive training system design // *New Solutions in Modern Technologies*. Kharkiv, Ukraine, № 79, 2006. С. 68-70.
2. *Bondarenko M.; Bilous N.; Shubin I.* The Ukrainian e-Learning Region // *In Proceedings of 10-th International LInE Conference New Partnerships and Lifelong Learning*. Helsinki, Finland, 2008. P. 88-92.
3. *Горбач Т.В., Святкин Я.В., Шубин И.Ю.* Представление и классификация неструктурированных данных в адаптивных обучающих мультимедиа-системах на основе метода компараторной идентификации // *Проблемы информационных технологий*.-- Херсон: Херсонский национальный технический университет. 2009. № 1 (005). С. 21-25.
4. *Горбач Т.В., Святкин Я.В., Шубин И.Ю.* Методы реализации адаптивной гипермедиа в обучающих системах // *Вестник Херсонского национального технического университета*. Херсон: Херсонский национальный технический университет. 2010. № 2 (38). С. 503-508.