

Автоматизации построения тестовых заданий в системах дистанционного обучения на основе понятийно-тезисной модели

Титенко Сергей Владимирович
к.т.н., доцент кафедры АПЭПС ТЭФ,
Национальный технический университет Украины
«Киевский политехнический институт»,
Украина, 03056, г.Киев-56, проспект Победы, 37
lab@setlab.net, www.setlab.net

АННОТАЦИЯ

Работа посвящена описанию понятийно-тезисной модели как средства для автоматизации тестирования. Представлено формальное описание модели и алгоритм построения тестового задания.

The work is devoted to description of concept-thesis model as a tool to testing automation. Formal description of the model and its algorithm are presented.

Ключевые слова

автоматизация тестирования, модель контента, индивидуальная учебная среда, онтология, понятийно-тезисная модель
testing automation, content management, content model, individual learning environment, ontology, concept-thesis model

Введение

Общество остро ощущает потребность в распространении учебных процессов в самых различных сферах жизни. В последнее время образовательные процессы, выходя за пределы образовательных учреждений, активно применяются в таких сферах как бизнес, корпоративное образование, повышение квалификации работников в разных отраслях.

Наиболее удобной формой обучения здесь является дистанционная. Одной из ключевых составляющих дистанционного образования является контроль и диагностика знаний, реализация которых в условиях отсутствия непосредственного контакта между участниками учебного процесса вызывают определенные технические и организационные трудности. Основной формой дистанционной проверки знаний является компьютерное тестирование. Ученые и специалисты в области электронного образования и информационных технологий внесли значительный вклад в разработку различных методов реализации тестирования. В работах [1, 2] описаны способы и стратегии организации компьютерного контроля, которые классифицируются по степени адаптивности, применению моделей студента и учебного материала, цели тестирования и другим критериям. В работе [3] представлен детальный обзор жизненного цикла тестовых заданий в образовательной

Web-системе, который включает такие стадии как подготовка, представление и оценка. Как правило, стадия подготовки тестовых заданий остается слабо автоматизированной и требует непосредственной работы преподавателя.

В то время, как многие исследования в области компьютерного контроля знаний сосредоточены на вопросах валидности и надежности тестов [4], вопросы формирования самого банка задач в большинстве случаев остаются исключительно прерогативой преподавателя, работающего без использования интеллектуальных средств автоматизации данного процесса. Действительно, попытки автоматизировать формирование тестовых заданий наталкиваются на область искусственного интеллекта и на проблемы формализации знаний и их дальнейшего использования для генерации тестов.

Традиционный подход к созданию средств тестирования фактически представляет собой компьютеризацию ручного тестирования. Суть такого подхода состоит в использовании информационно-коммуникационных технологий вместо бумажной работы, что дает дополнительные возможности управления формированием тестов из банка созданных задач и автоматической проверки результатов. Одновременно и преимуществом, и недостатком данного подхода является ручное, неавтоматическое создание заданий. Задания, профессионально разработанные экспертом, имеют высокое качество и понятность. В случае конкретных статических тестов имеют место широкие возможности для сосредоточения на вопросах валидности и надежности. Однако существенным недостатком подхода является высокая трудоемкость самого процесса по формированию тестовых заданий. Эта проблема усугубляется задачей защиты от недобросовестного прохождения теста, что предполагает наличие крупного банка задач и динамическую композицию на его основе индивидуального теста для предупреждения «списывания». Кроме того, преподаватель, создающий тестовые задания, несет педагогическую ответственность за адекватность тестового контроля учебному контенту. Поэтому от разработчика тестов требуются не только экспертные знания предметной области, но также глубокий анализ учебного материала, который предоставляется студентам дистанционной формы обучения как основной источник для овладения знаниями курса.

Постановка задачи

Целью работы является анализ существующих методов автоматизации построения тестовых заданий и разработка модели, которая позволила бы устранять выявленные в ходе анализа недостатки и организовать автоматизированное тестирование с учетом следующих требований:

- простота создания базы знаний, которая послужит основой для генерации тестовых заданий;
- возможность построения тестовых заданий разных типов;
- лексическая ясность тестовых заданий;
- возможность гибкого построения тестов в зависимости от необходимого участка учебного контента, цели и стратегии тестирования;
- возможность использования результатов тестирования для адаптивной настройки индивидуального учебного процесса;
- возможность интеграции созданной модели с моделями контента систем дистанционного обучения [5-7].

Методы генерации тестовых заданий

С целью интенсификации и упрощения процессов подготовки дистанционных учебных курсов с функциями тестирования учеными были предложены различные подходы к автоматизации создания тестовых заданий. Одним из перспективных и сравнительно несложных в реализации является подход параметризованных тестов [8-10]. Суть подхода заключается в представлении разным студентам шаблонного задания, отличающегося определенными параметрами, которые генерируются автоматически. Ответ вводится в открытой форме. Таким образом каждый студент получает индивидуальное задание, а система по определенной формуле или алгоритму, подставляя параметры, получает правильный ответ для дальнейшей проверки ответа, введенного студентом. Недостатком подхода является его узкая предметная направленность. Так, параметризованные тесты хорошо подходят для контроля практических навыков в точных науках [9], а также программировании [10], однако не подходят для проверки теоретических знаний, а также контроля в гуманитарных науках.

Определенное распространение в исследованиях автоматизации контроля знаний получил подход с применением семантических сетей для автоматизации построения тестов [11,12]. Краеугольным камнем семантических сетей являются так называемые триады [11-13] «сущность 1» – «отношение» – «сущность 2». Например, есть такие сущности «процедура» и «программа». В таком случае между ними можно установить отношение типа «является частью». Тогда получаем: сущность «процедура» «является частью» сущности «программа». Задания теста строятся путем опущения одного из звеньев триады и постановкой вопроса о недостающем звене. Преимуществом данного подхода является способность системы рассуждать знаниями предметной области. Недостаток заключается в больших затратах при составлении завершенной целостной семантической сети, которая бы корректно отражала данную предметную область изучения. Еще одним недостатком подхода является лингвистическая непонятность и порой нецелесообразность задач, которые генерируются. Так, на основе семантической сети часто задаются вопросы о таких особенностях объектов предметной области, которые не имеют в данном учебном контексте педагогической ценности. Подобные недостатки возникают на основе проблемы, характерной для классических моделей знаний искусственного интеллекта, которую можно назвать проблемой «всеобразованности». Так, требованием семантических сетей является полная формализация предметной области, тогда как изучение материала происходит в определенном контексте, поэтому при генерации задач в тест часто попадают вопросы, ответ на которые не свидетельствует о достижении или недостатке в обучении студента. Требование всеобразованности или всеобъемлющей формализации требует больших усилий для построения адекватной модели, при этом часто не обеспечивает должной педагогической целесообразности в процессе ее использования для образовательных целей и тестирования.

Эти и другие проблемы присущи для случаев, когда делается попытка применить классические модели искусственного интеллекта (ИИ) для образовательных задач. Несмотря на то, что как в моделях ИИ, так и в учебных системах, объектом моделирования являются знания, все же принципиальной разницей здесь является то, что в первом случае целью моделирования является предоставление системе способности «знать» определенную предметную область, чтобы решать некоторые прикладные задачи, тогда как во втором случае целью моделирования является предоставление системе способности «обучать» человека знаниям определенной предметной области [14]. Такая принципиальная разница в постановке задач требует разработки специальных моделей формализации знаний для образования и организации именно на их основе автоматизированного

тестирования. Основным средством передачи учебной информации студенту в дистанционном обучении является текст, в нем в понятной для обучения форме представлены знания. Исходя из этого, видится целесообразным положить в основу модели именно образовательный текст, сосредоточившись на методах его формализации с целью автоматизации построения тестовых заданий.

Понятийно-тезисная модель формализации дидактического текста

Понятийно-тезисная модель (ПТМ) формализации дидактического текста разрабатывается на стыке многих научных направлений, среди которых следующие: инженерия знаний как направление искусственного интеллекта; педагогика, а именно ее раздел – дидактика, которая раскрывает правила преподавания; интерпретация (герменевтика, экзегетика), которая изучает правила толкования текстов [15, 16]; лингвистика и ее раздел семантика, изучающие закономерности естественного языка и проблемы, связанные с содержанием, значением и интерпретацией лексических единиц; инструментом реализации служат технологии разработки Web-систем.

В образовательном процессе можно выделить следующие четыре составляющие: передача учебной информации, социальная составляющая, включающая общение между участниками образовательного процесса, практические занятия и контроль знаний. Самая первая миссия дистанционного образования – упрощение и оптимизация передачи учебной информации от носителя знаний к студенту. В традиционных формах обучения базовой формой передачи информации является вербальная – например, преподаватель читает лекцию. Вместо того в дистанционном, удаленном от преподавателя, обучении основной формой передачи учебной информации становится текст, а точнее гипертекст и мультимедиа содержимое. То есть, основным транспортным средством или «протоколом» передачи знаний является текст, в котором содержится учебная информация. Этот очевидный факт является основой понятийно-тезисной модели как средства представления знаний.

Краеугольным камнем структуры ПТМ является такая сущность, как понятие, предмет обсуждения, некоторый объект из предметной области, о котором в материале есть знание. Понятие указывает на некоторый объект из предметной области, о котором идет речь, и который представляется для изучения студенту. Например, в курсе «Алгоритмические языки программирования» можно выделить такие понятия: «процедура», «цикл», «программа», «переменная», «жизненный цикл программы» и другие. Для курса «Программирование в среде Delphi» можно было бы выделить следующие понятия: «объект», «событие», «класс», «форма», «компонент TEdit» и другие. Множество понятий в системе обозначается следующим образом:

$$C = \{c_1, \dots, c_{n1}\}.$$

Для представления знаний о понятии ПТМ содержит специальные структурные элементы – тезы. Теза – это сведение о понятии. Тезу можно сравнить с признаком, характеристикой или с любым утверждением, которое является истинным для данного понятия. В то время как понятие лишь указывают на предмет, о котором идет речь в тексте, теза является смысловым или описательным наполнением базы знаний. От полноты наборов таких утверждений зависит полнота базы знаний, а, следовательно, и способность обучающей системы строить эффективные задания теста. Формально теза представляет собой одно или несколько предложений, в которых речь идет непосредственно о некотором понятии, однако само понятие здесь словарно не фигурирует. С точки зрения лингвистики теза, как правило, является предложением, из которого удалено подлежащее. Приведем

примеры: теза о понятии «процедура» – «позволяет разбить программу на подпрограммы»; теза о понятии «класс» – «может иметь в своей структуре не только поля-свойства, но и методы, т.е. функции и процедуры» и другие. Множество тез в системе обозначается следующим образом:

$$T = \{t_1, \dots, t_{n_2}\}.$$

Понятие и тезы в совокупности будем называть ПТ-элементами. Каждая теза относится к одному понятию. Эта связь задается отношением:

$$CT: T \rightarrow C.$$

В свою очередь каждое понятие может иметь любое количество тез, что описывается соотношением:

$$TC: C \rightarrow 2^T.$$

Как отмечалось выше, основным носителем знаний считается учебный материал. Поэтому выделение конкретных семантических единиц тесно связано с процессом манипулирования учебным материалом. При подготовке методических материалов дистанционного курса весь материал, как правило, делится на структурные элементы контента. Понятие элемента контента является очень важным в ПТМ. С его помощью обеспечивается связь семантических элементов с учебным материалом. В данной работе мы не останавливаемся на конкретной реализации общей структуры хранения и представления учебного материала, более того, ПТМ допускает различные реализации этой структуры, что позволяет интегрировать ПТМ с уже существующими системами дистанционного обучения. Таким образом, в комплексе с другими составляющими ПТМ является одним из компонентов образовательной системы [17, 18], которая предусматривает структурирование учебного материала с помощью модели контента [5-7]. Обозначим множество элементов или страниц учебного контента:

$$V = \{v_1, \dots, v_{n_3}\}.$$

Семантические элементы ПТМ выделяются непосредственно из текста учебного фрагмента. Сам процесс формирования понятийно-тезисной базы фактически является осмысленным чтением учебного текста вместе с несложными манипуляциями к нему. Так, исследуя учебный материал, преподаватель выделяет непосредственно из текста и добавляет в БЗ важные учебные понятия и их тезы. В результате каждый фрагмент v_i может стать источником произвольного количества тез t_j , что задается отображением:

$$TV: V \rightarrow 2^T.$$

Каждая t_j , в свою очередь, касается одного учебного фрагмента v_i :

$$VT: T \rightarrow V.$$

Так, тезы относятся только к одному учебному фрагменту, из которого они были получены, тогда как понятия могут относиться ко многим учебным фрагментам. Связь между понятиями и учебным материалом обеспечивается опосредованно через тезы: понятие – теза – учебный материал. Понятия, относящиеся к данному учебному участку, определяются оператором:

$$CV(v)=\{c: TV(v) \cap TC(c) \neq 0\}.$$

Соответственно учебный материал, к которому относится данное понятие, определяется оператором:

$$VC(c)=\{v: TV(v) \cap TC(c) \neq 0\}$$

Схематическое изображение участка ПТМ в соотношении с учебным материалом показано на рис.1.

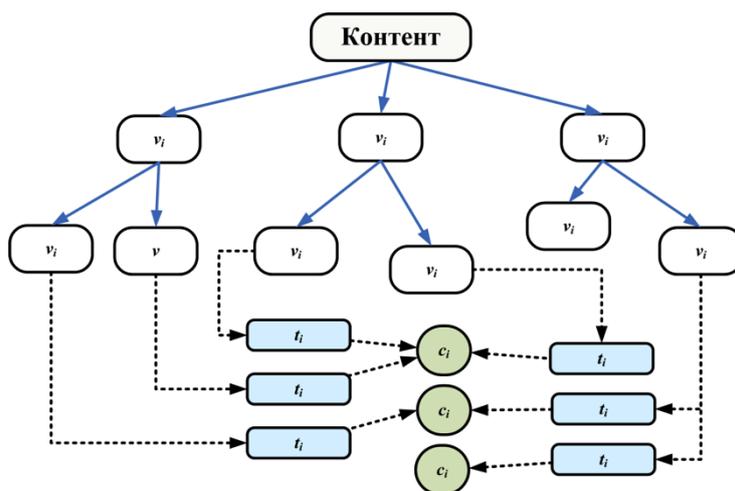


Рис.1. Участок ПТМ в соотношении с учебным материалом.

Классификация тез и понятий

Понятия и тезы могут быть отнесены к определенным классам. Эта классификация служит для сохранения в БЗ информации о смысловом или лексическом характере того или иного понятия или тезы. Классы понятий и тез используются в алгоритмах построения тестовых заданий. Базовый набор классов тез описывается множеством:

$$TClasses = \{tDefinition, tDestination, tEssence, tSyntax, tGeneral, tList, tCode, tAttaching\}$$

ПТМ предусматривает возможность расширения множества $TClasses$ пользовательскими классами тез, что позволяет расширять модель и настраивать ее для различных предметных областей. Дадим описание имеющихся классов в $TClasses$:

- $tDefinition$ обозначает тезу-определение;
- $tDestination$ обозначает тезу-назначение. Теза-назначение содержит информацию о назначении или роль того или иного понятия;
- $tEssence$ описывает так называемую тезу-сущность, которая служит для передачи некоторой принципиальной характеристики понятия;

- *tSyntax* обозначает тезу-синтаксис, характерную для предметной области программирования. Теза-синтаксис содержит синтаксис некоторой конструкции или команды языка программирования;
- *tGeneral* обозначает общую тезу, которая представляет ведомость общего характера о некотором понятии. Этот класс следует применять в случае, когда тезу нельзя отнести к другим классам;
- *tList* обозначает тезу-список, служащую для выделения из текста некоторого списка объектов или компетент, которые подают содержание или состав соответствующего понятия;
- *tCode* обозначает тезу, которая представляет описание программного кода, содержащегося в соответствующем понятии;
- *tAttaching* обозначает служебную тезу-прикрепление, которая указывает на то, что о соответствующем понятии говорится во фрагменте текста.

Отнесение тезы к определенному классу происходит благодаря отображению:

$$TClass = T \rightarrow TClasses$$

Понятия, так же, как и тезы классифицируются с целью отнесения их к определенной смысловой группе. Множество классов понятий:

$$CClasses = \{cGeneral, cCode\}$$

Базовый набор классов содержит два класса: *cGeneral* – для обозначения понятия общего типа; *cCode* – для обозначения понятия, которое дает программный код. ПТМ предусматривает возможность добавления классов для настройки представления различных предметных областей. Понятия классифицируются с помощью отображения:

$$CClass = C \rightarrow CClasses$$

Возможности применения ПТМ для генерации тестовых заданий

Понятийно-тезисная модель формализации дидактического текста служит основой для генерации тестовых заданий. Далее приводятся общие возможности и направления использования ПТМ для генерации тестовых заданий разных типов. Отметим распространенные типы задач, которые могут применяться при контроле знаний путем тестирования и имеющих перспективное значение для разработанной семантической модели.

1. Простейшим типом является вопрос, что предполагает ответ типа «Истина/Ошибка».

2. Самым распространенным видом заданий в тестах есть задания множественного выбора: «один из нескольких» и «несколько из нескольких».

3. Более сложным является задача со свободной формой ответа: ответ на вопрос студент должен ввести вручную.

4. Еще один вариант – задача сопоставления. Студенту подается два набора некоторых элементов, и он должен сопоставить каждый вариант одного набора с соответствующим по его мнению вариантом с другого набора элементов.

5. Отдельно выделим еще один достаточно специфический тип заданий – задание по определению приоритетности. Суть задания заключается в том, чтобы расставить определенные элементы в правильном порядке или порядке их приоритетности.

Сосредоточимся на путях и алгоритмах построения выше перечисленных типов задач на основе ПТМ.

1. Вопросы **первого типа**, требующие ответа типа «Истина/Ошибка», строятся следующим образом: с ПТ-базы добывается соответствующая пара понятие-теза и студенту ставится вопрос, верно ли данное утверждение:

Отвечает ли данное утверждение понятию <понятие>?

<теза>

Да Нет

Алгоритм последующей проверки ответа очевиден – если полученная теза действительно касается данного понятия, верным ответом будет «Да» («Истина»).

2. Тесты **второго типа** «вопрос – варианты ответов» реализовывать относительно не сложно. Есть несколько путей построения таких заданий: (1) в основе вопроса лежит понятие, в основе вариантов ответов – тезы, (2) в основе вопроса лежит теза, в основе вариантов ответов – понятие; некоторые другие вариации.

Раскроем указанные пути.

1) В основе вопроса лежит понятие, в основе вариантов ответов – тезы. Самым общим здесь будет задание такого вида:

Укажите утверждение, касающееся данного понятия <имя понятия>

<перечень утверждений>

В зависимости от класса понятия и тезы вопрос может звучать иначе, например:

«Для чего предназначено понятие ...», если тезы, которые служат вариантами ответов, относятся к классу «назначения»;

«Что есть понятие ...», если тезы класса «определения»;

2) В основе вопроса лежит теза, в основе вариантов ответов – понятие. То есть, в этом методе в основе вопроса лежит сведение о понятии. Студент должен из набора понятий выбрать то, о котором идет речь в этом сведении. Так же, как и в предыдущем способе, задание может иметь как общий характер:

Укажите понятие, о котором идет речь в утверждении

<Утверждение>

<Перечень понятий>,

так и конкретный, в зависимости от класса понятия и тезы: *«Укажите понятие, которое определено ниже ...»*, если теза типа «определение»; *«Какое понятие имеет такое назначение ...»*, если теза имеет тип «назначение».

3) На основе ПТ-базы можно построить и другие варианты заданий типа «вопрос – варианты ответов».

Одним из возможных направлений здесь является задача по определению важнейших элементов учебного материала. Принцип такой: перед студентом ставится задача определить, например, важнейшее понятие в данной лекции из перечня понятий. Важнейшее понятие системой определяется на основе количества тез этого понятия на данном участке материала с учетом веса важности.

Еще одно направление – использование возможностей наследования и отношений между понятиями. Задания строятся по принципу «понятие - набор понятий». В качестве вопроса ставится лексическая форма отношения между понятиями и само понятие, например: *«Укажите родительское понятие ...»*. В качестве вариантов ответов – набор понятий, среди которых есть понятие, которое удовлетворяет условие задачи.

3. Рассмотрим тестовые задания **третьего типа**: вопрос – текстовый ответ вводится студентом. С организационно-методологической точки зрения этот тип несколько подобен первому, но по уровню учебно-методической сложности он тяжелее для студента.

Для этого вида теста подходят задачи, в которых в качестве ответа выступает понятие. Самый общий вид задачи будет построен следующим образом: указывается некоторая теза и студенту предлагается ввести понятие, к которому относится данная теза. Например:

Общий вариант:

<Теза>

О каком понятии идет речь?

На основе определения:

<Теза-определение>

Что это за понятие?

На основе назначения:

<Теза-назначение>

Назначение какого понятия приведено выше?

И т.д.

Кроме этого, можно использовать сведения о наследовании и отношениях между понятиями. Так на базе наследования можно построить задание типа:

Укажите родительское понятие к понятию <Название понятия>.

Характерной особенностью, с точки зрения реализации, является требование дополнительной подсистемы для сравнения текстового ответа с эталоном (с верным ответом) и оценки его правильности. Этот алгоритм должен предусматривать возможные орфографические ошибки в тексте, которые в определенной мере не должны влиять на правильность ответов (если только это не обучение по дисциплинам лингвистического цикла). Если речь идет об ответах, состоящих из многих слов, тогда нужны соответствующие алгоритмы обработки. Следующим витком развития системы проверки текстового ответа может стать создание лингвистической подсистемы, которая сможет применять такие знания как синонимические ряды, разного рода словоформы, спряжения, времена для глаголов и т.п.

4. Четвертый вариант – задание сопоставления. Студенту подается два набора некоторых элементов и он должен сопоставить каждый вариант одного набора с соответствующим, по его мнению, вариантом другого набора элементов.

Очевидным путем реализации такого типа задач является такой: первый набор элементов - это понятие, другой – тезы к этим понятиям. Между двумя наборами должна существовать взаимно однозначная связь – каждому понятию соответствует лишь одна теза из набора тез. Студент с помощью соответствующих средств интерфейса осуществляет сопоставление элементов.

Важной составляющей такой задачи является методика оценки ответа студента. Самый простой путь – это оценка типа «зачет» («незачет»), без дифференцирования. Однако, для данного типа задач очевидна потребность в градации оценки, ведь возможен случай, когда студент сопоставит часть пар верно, а часть – нет. В таком подходе простым является путь оценки на основе количества верных ответов, тогда процентом правильности ответа будет процент количества правильных сопоставлений. Более гибкий подход – использование веса, когда для каждой пары формируется своя весовая оценка, то есть доля в общей оценке, при проверке учитывается в общей оценке. Весовая оценка пар строится на базе степени важности элементов (как понятия, так и тезы), упоминаемых на этапе формирования базы знаний.

5. Пятый довольно специфический тип заданий – задачи по определению приоритетности. Суть задания заключается в том, чтобы расставить определенные элементы в правильном порядке, порядке их приоритетности, или в том, чтобы определенным образом, выделить элементы, имеющие больший вес.

По структуре такое задание временами аналогичное заданиям, описанным выше. Отличием является учебно-методический смысл, заложенный в нем. Вопросы

строятся на основе вторичных знаний, полученных с понятийно-тезисной базы посредством использования сведений о степени важности ПТ-элементов. Каждый ПТ-элемент имеет свой уровень важности, который может быть указан на этапе формирования БЗ. Разные понятия курса имеют разный уровень важности в структуре знаний курса. Итак, понимание некоторых понятий является абсолютно необходимым для усвоения курса, остальные играют вспомогательную, такую, что расширяет кругозор, роль. То же касается утверждений о понятии.

Таким образом, задача этого типа содержит аналитический характер и требует от студента глубокого понимания предмета. Приблизительные вопросы (задания) могут иметь такой вид:

«Расставьте в порядке важности понятия курса ...»;

«Какое понятие важнее в этом разделе ...»;

«Укажите ключевые понятия темы»;

«Укажите утверждение, выражающее главное свойство понятия ...»;

Эффективность заданий зависит от адекватности заложенных оценок важности учебных элементов модели.

Структура теста и тестового задания

Здесь приводится формальная модель и алгоритм теста на основе ПТМ, который состоит из заданий множественного выбора.

Тест и его структура. В основе генерации тестов в ПТМ лежит информация о понятиях, тезах, учебном материале и их взаимосвязях. Множество тестов задается множеством:

$$Test = \{test_i\}$$

Входным параметром теста есть контрольная область учебного контента, то есть набор фрагментов учебного материала, по которым будет происходить тестирование. Контрольная область контента задается отображением множества тестов на множество учебных фрагментов:

$$Vtrg: Test \rightarrow 2^V$$

Прежде, чем начнется построение тестовых заданий, осуществляется определение так называемой ресурсной области контента, которая служит дополнительным источником ПТ-элементов, которые могут использоваться как альтернативные ответы для тестовых заданий. Ресурсная область контента задается отображением:

$$Vres: Test \rightarrow 2^V$$

В простейшем случае ресурсная область контента $Vres(test)$ совпадает с контрольной областью $Vtrg(test)$. Однако, учитывая то, что строительным материалом тестовых заданий является ПТ-элементы, возможны случаи, когда семантических единиц ПТМ будет недостаточно для построения необходимого количества заданий теста, кроме того, использование ПТ-элементов из других источников может улучшить качество тестовых заданий. В таких случаях ресурсная область контента $Vres(test)$ определяется на основе поиска учебных фрагментов, материал которых касается той же предметной области, что и контрольный материал

$Vtrg(test)$. Подобные алгоритмы зависят от структуры и возможностей модели учебного контента. Определение предметной области в иерархически-сетевой модели образовательного контента Tree-Net, с которой интегрируется ПТМ, рассмотрены в предыдущих работах [6].

Тестовое задание. ПТМ позволяет генерировать задания теста на основе информации о понятии, тезах и их связи с учебным материалом. Представим описание технологии автоматизированного создания тестовых заданий. Множество тестовых заданий обозначается следующим образом:

$$Task = \{task_i\}.$$

Связь тестов с заданиями задается отображением:

$$TestTasks: Test \rightarrow 2^{Task}.$$

В свою очередь, каждое задание связывается с контрольной ПТ парой, то есть таким понятием и его тезой, которые лягут в основу этого тестового задания. Связь задачи с контрольной ПТ парой задается отображением:

$$TaskCT: Task \rightarrow CT.$$

Существует набор шаблонов для тестовых заданий $TTempl = \{TTempli\}$, которые, кроме того, могут расширяться пользователем, что позволяет дополнять различные предметные области новыми типами тестовых заданий. Шаблоны, которые применяются для тестовых заданий, задаются отображением:

$$TaskTempl: Task \rightarrow TTempl.$$

Каждое задание связывается с набором семантических элементов, которые используются как варианты альтернативных ответов. В зависимости от шаблона в качестве альтернативных ответов могут выступать или тезы, или понятия. Связь между задачами и альтернативными вариантами задается отображением:

$$TaskAItems: Task \rightarrow 2^C \cup 2^T$$

Шаблон тестового задания. Шаблоны имеют целью описать способ построения тестовых заданий на основе ПТ-элементов. Возможность добавлять новые шаблоны позволяет совершенствовать систему не только на этапе проектирования, но и на этапе ее использования для настройки алгоритмов генерации тестов для различных предметных областей. Представим структурные связи и особенности использования шаблонов для построения тестов на основе ПТМ. Шаблоны задач делятся на два типа: 1) в основе вопроса лежит понятие, а в основе вариантов ответов – тезы, 2) в основе вопроса лежит теза, а в основе вариантов ответов – понятия. Таким образом, типизация запросного ПТ-элемента задается отображением:

$$TQEntity: TTempl \rightarrow CTEntity, \text{ де } CTEntity = \{Concept, Thesis\}.$$

Для задания шаблонного текста, который отвечает за вопросы тестового задания, задается отображение шаблонов на соответствующие элементы-образцы вопросов:

$$TQStr: TTempl \rightarrow QStr, \text{ де } QStr = \{qStr_i\} - \text{множество текстовых шаблонов вопросов}$$

Место в тексте $qStri$, куда должен быть вставлен текст запросного ПТ-элемента, обозначается специальным образом, а именно символами «###». Для обозначения классов ПТ-элементов, которые могут быть применены для тестового задания каждого из шаблонов, служит отображение:

$$TQClasses: TTempl \rightarrow 2^{CClasses} \cup 2^{TClasses}$$

Как альтернатива $TQClasses$ может быть применено указание классов, которые запрещено использовать в задании, что означает разрешение на использование всех других классов ПТ-элементов. Для этого служит следующее отношение:

$$TQNotClasses: TTempl \rightarrow 2^{CClasses} \cup 2^{TClasses}$$

Для аналогичного указания классов для ПТ-элементов, служащих как варианты ответов, используются отображения $TAClasses$ и $TANotClasses$:

$$\begin{aligned} TAClasses: TTempl &\rightarrow 2^{CClasses} \cup 2^{TClasses}, \\ TANotClasses: TTempl &\rightarrow 2^{CClasses} \cup 2^{TClasses} \end{aligned}$$

Генерация тестового задания

Представим последовательность построения и визуализации тестового задания $taskk$ теста на основе ПТ-элементов, их связи с учебными материалами и шаблонов тестовых заданий. Обозначим как $testk$ тест, в рамках которого строится задача:

$$task_k \in TestTasks(test_k).$$

1. Выбор контрольной ПТ-пары. Случайным образом выбирается контрольная ПТ-пара из контрольной области учебного контента $TaskCT(task_k) = (t_k, c_k)$:

$$TaskCT(task_k) = (t_k, c_k): (t_k, c_k) \in CT \wedge \forall t_k \in Vtrg(test_k).$$

2. Поиск допустимых шаблонов задач. На основе классов контрольного понятия c_k и тезы t_k находится множество всех возможных вариантов шаблонов TT . Во-первых, находятся все возможные шаблоны для случая, когда в основе тестового вопроса стоит понятие, $TQEntity(tt) = Concept$. Обозначим это множество шаблонов задач TT' , $TT' \subseteq TTempl$:

$$\begin{aligned} TT' = \{ tt: TQEntity(tt) = Concept \wedge (CClass(c_k) \in TQClasses(tt) \vee TQClasses(tt) = 0) \wedge \\ CClass(c_k) \notin TQNotClasses(tt) \wedge (TClass(t_k) \in TAClasses(tt) \vee TAClasses(tt) = 0) \wedge \\ TClass(t_k) \notin TANotClasses(tt) \}. \end{aligned}$$

Аналогично находятся все возможные шаблоны для случая, когда в основе тестового вопроса стоит теза, $TQEntity(tt) = Thesis$. Это множество шаблонов задач обозначим TT'' , $TT'' \subseteq TTempl$:

$$TT'' = \{ tt: TQEntity(tt) = Thesis \wedge (TClass(t_k) \in TQClasses(tt) \vee TQClasses(tt) = 0) \wedge TClass(t_k) \notin TQNotClasses(tt) \wedge (CClass(c_k) \in TAClasses(tt) \vee TAClasses(tt) = 0) \wedge CClass(c_k) \notin TANotClasses(tt) \}$$

Таким образом совокупное множество всех возможных шаблонов задач для задачи с контрольной ПТ-парой $TaskCT(task_k)$ объединит оба случая:

$$TT = TT' \cup TT''$$

3. Выбор шаблона задачи. С множества шаблонов TT случайным образом выбирается один шаблон tt_k , что и будет служить шаблоном задачи: $TaskTempl(task_k) = tt_k$ для генерации и визуализации тестового задания $task_k$.

4. Поиск альтернативных вариантов ответов. В соответствии с выбранным шаблоном осуществляется поиск ПТ-элементов, которые могут служить как альтернативные варианты ответа. В случае, когда $TQEntity(tt_k) = Concept$, возможные альтернативные варианты ответа выбираются из множества тез, класс которых соответствует параметрам шаблона $tt_k = TaskTempl(task_k)$:

$$TaskAItems' = \{ t: t \neq t_k \wedge CT(t) \neq c_k \wedge ((TClass(t) \in TAClasses(tt_k) \wedge TAClasses(tt_k) \neq 0) \vee (t \in T \wedge TAClasses(tt_k) = 0)) \wedge TClass(t) \notin TANotClasses(tt_k) \wedge VT(t) \in Vres(test_k) \}.$$

В случае, когда $TQEntity(tt_k) = Thesis$, возможные альтернативные варианты ответа выбираются из множества понятий, класс которых соответствует параметрам шаблона $tt_k = TaskTempl(task_k)$:

$$TaskAItems' = \{ c: c \neq c_k \wedge CT(t) \neq c_k \wedge ((CClass(c) \in TAClasses(tt_k) \wedge TAClasses(tt_k) \neq 0) \vee (c \in C \wedge TAClasses(tt_k) = 0)) \wedge CClass(c) \notin TANotClasses(tt_k) \wedge VC(c) \cap Vres(test_k) \neq 0 \}.$$

Из полученного множества случайным образом выбирается подмножество $TaskAItems(task_k)$, мощность которого зависит от количества вариантов ответов, которые должны войти в задания.

5. Визуализация тестового задания. Когда все семантические данные готовы, осуществляется отображение тестового задания пользователю.

Вопрос строится на основе текстового шаблона $TQStr(task_k)$. В предусмотренное шаблоном место вставляется текст запросного элемента контрольной ПТ-пары $TaskCT(task_k) = (c_k, t_k)$. В случае, когда в основе вопроса стоит понятие, т.е. $TQEntity(tt_k) = Concept$, в вопросе будет фигурировать текст понятия c_k , в противном случае – текст тезы t_k .

Список вариантов ответов формируется из набора неправильных ответов $TaskAItems(task_k)$ и правильного ответа, получаемого из контрольной ПТ-пары. Список вариантов ответов сортируется случайным образом и вместе с вопросом подается пользователю в качестве тестового задания. Процесс генерации тестового задания показан на рис.2.

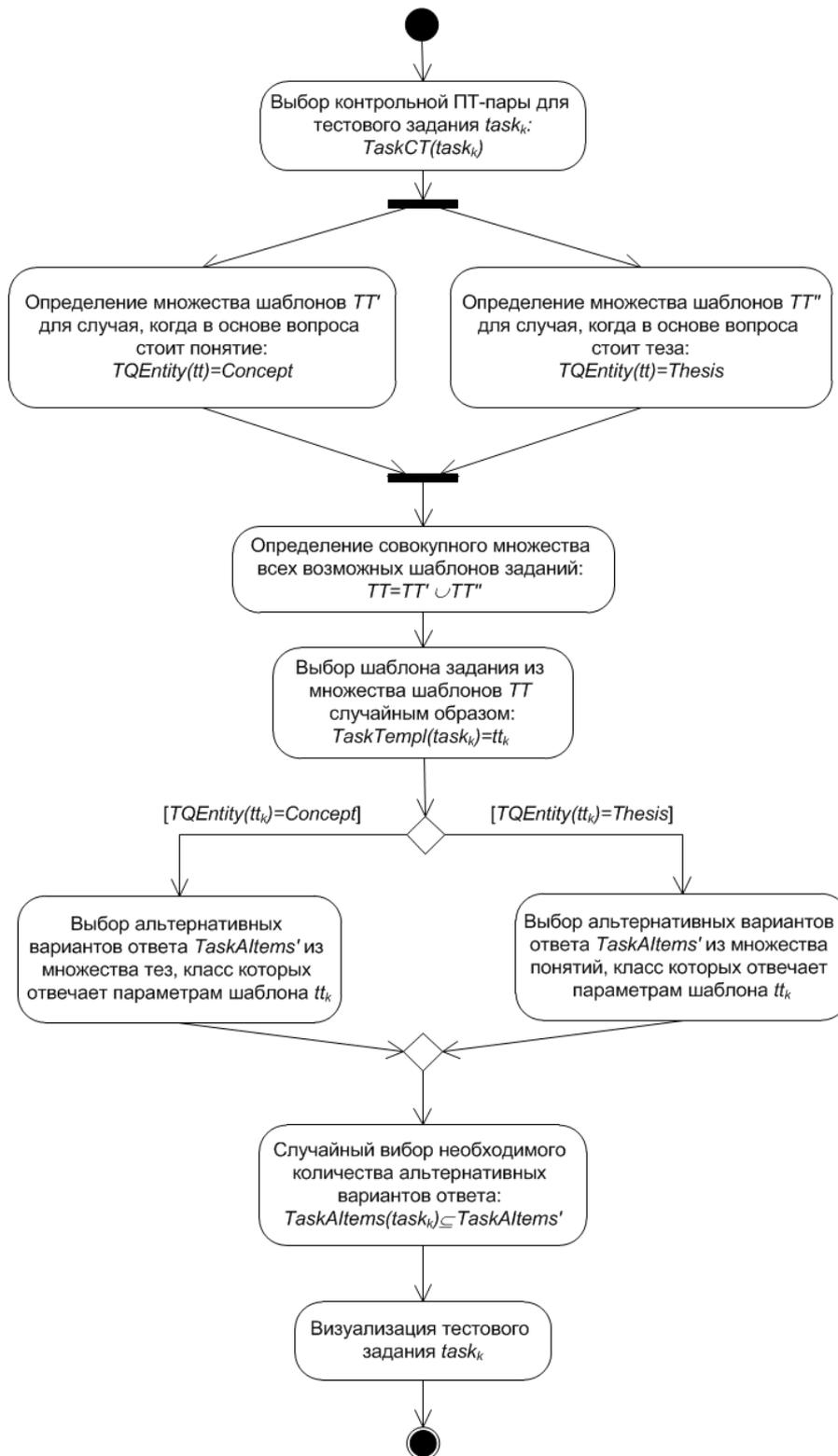


Рис.2. Генерация тестового задания. Диаграмма деятельности в нотации UML.

После получения ответа студента на сформированное задание происходит анализ правильности путем сравнения ответа с истинным вариантом. В случае неверного ответа данные о контрольных понятиях c_k , теза t_k и участок учебного материала $VT(t_k)$ используются для информирования студента о пробелах в изучении соответствующего учебного материала и учебного понятия, а также для коррекции дальнейшего сценария обучения. Таким образом, после прохождения теста из набора автоматически сформированных тестовых заданий системой накапливается информация о достижениях студента и его пробелах в обучении. Выявленные пробелы могут стать сигналом для более детальной проверки знаний по соответствующему участку учебного материала или степени усвоения соответствующего учебного понятия. Все это позволяет организовывать тестирование, адаптивное относительно учебной темы, понятия или раздела. Схема использования понятийно-тезисного подхода для автоматизации тестирования изображена на рис.3.

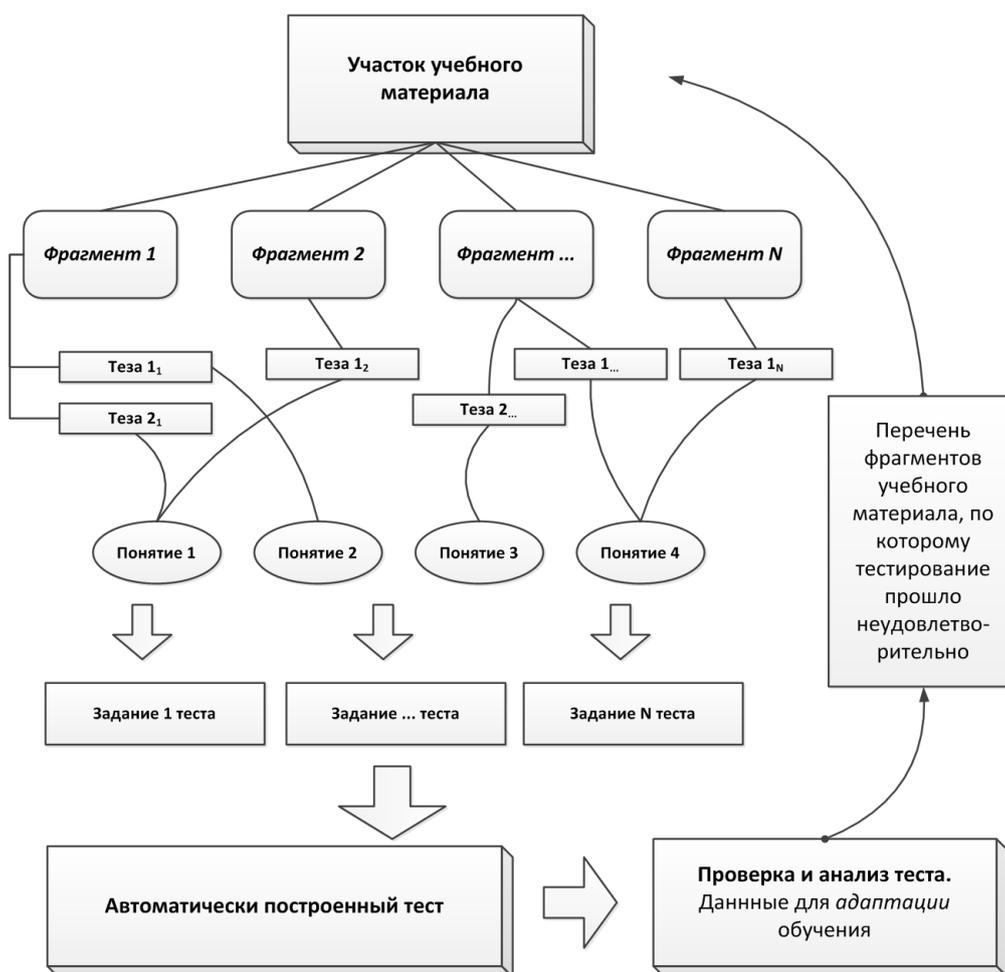


Рис.3. Схема использования понятийно-тезисного подхода для автоматизации тестирования.

Сравнение ПТМ с другими подходами построения тестовых заданий

Описанные подходы к построению тестовых заданий были реализованы и апробированы в открытом учебном процессе на портале дистанционного обучения [19]. Практика использования ПТМ позволяет сделать сравнительный анализ технологии ПТМ с традиционной технологией ручного построения тестов и технологией автоматизированного построения тестовых заданий на основе семантических сетей. Сравнительный анализ приводится в табл. 1.

Таблица 1. Сравнительный анализ технологий построения тестовых заданий.

Критерий		Технологии тестирования		
		Традиционный подход	Семантические сети	Понятийно-тезисная модель
1	Принцип работы	Компьютерная реализация традиционного тестирования. На основе педагогической технологии.	На основе базы знаний семантической сети. Центральной структурной идеей для формирования тестовых вопросов является триада: «понятия» - «отношение» - «понятия». Технология в рамках традиционной непедагогичной постановки задачи для ИИ.	На основе понятийно-тезисной базы. Центральной структурной идеей для формирования тестовых заданий есть связь «учебный материал» - «тезы» - «понятия». Модель носит педагогический характер с элементами применения инженерии знаний.
2	Способ формирования базы знаний или банка заданий	Ручным способом. Фактическое создание вопросов и вариантов ответов преподавателем.	Формирование базы знаний экспертом предметной области. Определение понятий и связей между ними.	Формирование базы знаний путем выделения из текста учебного материала семантических единиц по несложной схеме.
3	Трудовые и временные затраты, необходимые для подготовки теста	Высокие. Прямо пропорциональные количеству необходимых вопросов. Трудно оптимизируются.	Необходимы большие затраты на формирование базы знаний. Однако формирование тестов происходит автоматически.	Сравнительно небольшие затраты. Генерация тестов происходит автоматически.
4	Необходимость привлечения экспертов для формирования тестов или БЗ	Нет необходимости. Формирования тестов может осуществлять преподаватель соответствующей дисциплины.	Кроме эксперта по данной предметной области требуется инженер по знаниям.	Нет необходимости. Процесс формирования тестов легкий для восприятия, его выполнение может осуществлять преподаватель соответствующей дисциплины.
5	Качество тестовых заданий и их лексическая ясность	Высокое, так как их формирование осуществляется непосредственно человеком.	Задания часто трудны для восприятия человеком. Это связано с тем, что семантические	Хотя качество и уступает задачам созданным вручную, однако оно достаточно высоко, так как вопросы строятся на

			сети создавались для «объяснения» знаний компьютеру, а поэтому плохо подходят для общения с человеком.	основе фрагментов учебного текста, созданного человеком.
6	Степень автоматизации генерации заданий теста	Фактически отсутствует. Используются готовые задачи.	Полная автоматизация на основе созданной семантической сети.	Полная автоматизация на основе семантических данных ПТМ и их связей с учебным материалом.
7	Соответствие теста учебным материалам, представленным для изучения студентам, в том числе в условиях реструктурирования курса	Соответствие теста учебному материалу закладывается на этапе создания тестовых заданий и зависит от конкретной системы. В случаях необходимости проводить тестирование по некоторым конкретным участкам учебного курса или его реструктурировании, появляется необходимость обогащения тестовых заданий мета-данными об учебном участке.	В связи с тем, что для создания семантической сети применяются знания о предметной области вообще, связь с конкретными учебными материалами должен быть установлен отдельно. При этом усложняется контроль соответствия тестового задания конкретному участку учебного материала.	Связь учебного материала и семантических данных заложены в самой модели, и он задается естественным образом во время формирования базы знаний. Это позволяет при реструктурировании учебного материала корректно генерировать соответствующие тестовые задания.
8	Возможность выработки рекомендаций для управления дальнейшим процессом обучения	В зависимости от степени интеграции с учебным материалом и наличия в тестовых заданиях мета-данных об учебном участке.	Рекомендации могут обращать внимание студента на конкретные понятия, которые были слабо усвоены им при изучении.	Благодаря глубокой интеграции с учебным материалом является возможность точно определять, какие участки учебного материала требуют повторения, а также обратить внимание студента на конкретные учебные понятия.

Заключение

Понятийно-тезисная модель является попыткой решить проблему несовместимости задач традиционного искусственного интеллекта и систем автоматизированного обучения путем формализации знаний с точки зрения обработки непосредственного учебного текста, который выступает первичным источником знаний. Формирование ПТ-базы происходит в процессе осмысленного чтения учебного текста и несложных манипуляций с пользовательским интерфейсом, что значительно упрощает процесс построения базы знаний.

ПТМ обеспечивает систему дистанционного обучения генератором тестовых заданий, который, благодаря уникальности каждого из построенных тестов, дает возможность свести к минимуму проблему недобросовестного тестирования, характерную для статических тестов или тестов с недостаточно большим банком задач. В то же время трудовые затраты на построение ПТ-базы меньше трудовых

затрат на построение достаточно большого банка заданий, и, соответственно, меньше затрат на построение достаточно полной семантической сети предметной области. В отличие от тестовых заданий, генерируемых на основе семантических сетей, задачи на основе ПТМ обладают лучшей лексической понятностью, что положительно влияет на качество контроля. Точная связь семантических данных с участками учебного материала позволяет использовать различные стратегии при формировании тестов и их оценке. Так, ПТМ позволяет строить тест по любой указанной области учебного контента, а также по результатам тестирования, позволяет точно определить те учебные фрагменты и учебные понятия, которые нуждаются в доработке студентом. Такие возможности могут быть использованы для дальнейшей адаптации индивидуального учебного процесса.

ПТМ может быть легко интегрирована с моделями контента систем дистанционного обучения, в частности ПТМ интегрируется с иерархически-сетевой моделью образовательного контента Tree-Net, что позволяет строить индивидуализированную обучающую среду с такими функциями, как генерация индивидуальных тестовых заданий и генерация индивидуального учебного курса [14, 15].

Практическая реализация ПТМ и автоматизированное тестирование осуществлены в рамках открытого портала дистанционного обучения [19]. Программная реализация содержит средства пополнения ПТ-базы на основе асинхронного Web-интерфейса и модуля генерации и анализа тестов на основе ПТМ.

Дальнейшие исследования должны быть сосредоточены на повышении качества тестовых заданий и совершенствовании методов и алгоритмов их генерации. Перспективным здесь является расширение базовой классификации тез и понятий и формирование на основе этой классификации новых шаблонов для заданий теста. Еще одним перспективным направлением является построение модели шаблона для тестового задания открытого типа. Текущие исследования ПТМ освещаются на авторском сайте [20].

Литература

1. Прокофьева Н. О. Стратегии и методы проведения контроля знаний в компьютерном обучении // *Educational Technology & Society* — 13 (1). 2010 — pp. 378-392 ISSN 1436-4522.
2. Прокофьева Н. О. Вопросы организации компьютерного контроля знаний // *Educational Technology & Society* — 13 (1). 2010 — pp. 433-440 ISSN 1436-4522
3. Brusilovsky, P. and Miller, P., Web-based testing for distance education. In: P. De Bra and J. Leggett (eds.) *Proceedings of WebNet'99, World Conference of the WWW and Internet, Honolulu, HI, Oct. 24-30, 1999, AACE*, pp. 149-154
4. Аванесов В.С. Композиция тестовых заданий. Учебная книга для преподавателей вузов, учителей школ, аспирантов и студентов пед. вузов. 2 изд., испр. и доп. М.: Адепт. 1998-217с.
5. Титенко С.В., Гагарин О.О. Ієрархічно-мережева модель управління освітнім контентом системи безперервного навчання // Системний аналіз та інформаційні технології: Матеріали X Міжнародної науково-технічної конференції (20-24 травня 2008р., Київ). – К.: НТУУ «КПІ», 2008 – 258 с.
6. Титенко С.В., Гагарин О.О. Модулювання області знань в системі безперервного навчання на основі інтеграції моделі контенту Tree-Net і понятійно-тезисної моделі // VIII міжнародна конференція «Інтелектуальний аналіз інформації ІАІ-2008», Київ, 14-17 мая 2008г.

- : Сб. тр./ Ред. кол. : С.В. Сирота (гл.ред.) и др. – К.: Просвіта, 2008. – С. 475-484.
7. Титенко С. В. Онтологически-ориентированная система управления контентом информационно-учебных Web-порталов / С. В. Титенко // *Educational Technology & Society* — 15 (3). 2012. — pp. 522-533. ISSN 1436-4522
 8. Pathak S., Brusilovsky P. Assessing Student Programming Knowledge with Webbased Dynamic Parameterized Quizzes. In Barker P and Rebelsky S (eds) *ED-MEDIA'2002 - World Conference on Educational Multimedia, Hypermedia and Telecommunications*, Denver, CO, June 24-29, 2002, pp. 1548-1553
 9. Левинская М.А. Автоматизированная генерация заданий по математике для контроля знаний учащихся // *Educational Technology & Society* - 5(4). 2002. - pp. 214-221. ISSN 1436-4522
 10. QuizPACK - Quizzes for Parameterized Assessment of C Knowledge <http://www.sis.pitt.edu/~taler/QuizPACK.html>
 11. Елизаренко Г.Н. Проектирование компьютерных курсов обучения: концепция, язык, структура. НТУУ «КПИ». Киев, 2001.
 12. Slavomir Stankov, Branko Žitko and Ani Grubišić. Ontology as a Foundation for Knowledge Evaluation in Intelligent E-learning Systems. *AIED'05 Workshop SW-EL'05: Applications of Semantic Web Technologies for E-Learning. Papers of 12th International Conference on Artificial Intelligence in Education (AIED 2005)*. Amsterdam, 2005. <http://hcs.science.uva.nl/AIED2005/W3proc.pdf>
 13. Berners-Lee T. "Spinning the Semantic Web: Bringing the World Wide Web to Its Full Potential", The MI Press, 2005
 14. Гагарін О.О., Гайдаржи В.І., Титенко С.В. Концептуальний підхід до подання знань в інтелектуальній освітній системі // Сучасні тенденції розвитку інформаційних технологій в науці, освіті та економіці: Матеріали Всеукраїнської науково-практичної конференції. 11-13 грудня 2006 р., м. Луганськ. – Луганськ: Альма-матер, 2006. – С.17-19
 15. Артур К. Как изучать Библию. Санкт-Петербург, 1998.
 16. Gordon D. Fee, Douglas Stuart. *How to Read the Bible For All its Worth: A Guide to Understanding the Bible*. Michigan, 1982.
 17. Gagarin A., Tytenko S. Complex model of educational hypermedia environment for ongoing learning // *Образование и виртуальность – 2007. Сборник научных трудов 11-й Международной конференции Украинской ассоциации дистанционного образования / Под общ. ред. В.А. Гребенюка, Др Киншука и В.В. Семенца.*– Харьков-Ялта: УАДО, 2007.– С. 140-145
 18. Гагарін О.О., Титенко С.В. Проблеми створення гіпертекстового навчаючого середовища // *Вісник Східноукраїнського національного університету імені Володимира Даля №4 (110) 2007 Ч.2* - Луганськ 2007 - С. 6-15.
 19. Портал знань – застосування ПТМ підходів для автоматизації Web-тестування <http://www.znannya.org>
 20. Лабораторія СЕТ – Віртуальна лабораторія новітніх інформаційних технологій. Дослідження в області дистанційного навчання <http://www.setlab.net>