А. М. Куртасов, А. Н. Швецов

ФГБОУ ВПО «Вологодский государственный технический университет», г. Вологда

ПРОГРАММА ГЕНЕРАЦИИ УЧЕБНЫХ ТЕСТОВ НА ОСНОВЕ СЕМАНТИЧЕСКОГО ПОДХОДА

АННОТАЦИЯ

В докладе предлагается подход к задаче генерации учебных тестов, основанный на технологии извлечения знаний из естественно-языковых текстов. Рассматривается проблема семантического анализа обрабатываемых текстов.

ВВЕДЕНИЕ

В настоящее время на рынке существует большое количество программных продуктов, предназначенных для компьютерной реализации учебного тестирования: *UniTest System*, «СИнТеЗ», «Прометей», *Moodle* и др. [1]. Практически все эти продукты обеспечивают широкие возможности для проведения тестирования и оценки результатов. Однако наиболее важная и сложная для выполнения задача – композиция тестовых заданий – до сих пор остается слабо автоматизированной. Тестовые вопросы и варианты ответов создаются вручную, и этот процесс отличается большой трудоемкостью.

В данном докладе развивается авторский подход к автоматизации рассматриваемой задачи, основанный на извлечении знаний из естественно-языковых текстов. В качестве таких текстов могут использоваться пособия и учебники по различным учебным дисциплинам. Исходные принципы подхода были заложены в нескольких ранее опубликованных работах [2], [3]. Основная идея подхода заключается в том, что из составляющих текст утвердительных предложений можно автоматически генерировать вопросы, которые затем будут отбираться, корректироваться и редактироваться преподавателем.

С целью практической апробации подхода в настоящее время разрабатывается программный продукт, позволяющий формировать тестовые задания для последующей их передачи в *LMS*-систему *Moodle* (http://moodle.org/), которая используется в качестве основного средства дистанционного обучения в Вологодском государственном техническом университете.

1. ОСНОВНЫЕ ПРИНЦИПЫ РАБОТЫ ПРОГРАММЫ

Основными средствами, реализующими данный подход, являются лингвистические процессоры, которые друг за другом обрабатывают входной текст. Вход одного процессора является выходом другого. Процессоры выполняют анализ текста на разных уровнях языка: графематический анализ (выделение предложений, слов, чисел, формул и т. д.); морфологический анализ (построение морфологической интерпретации слов входного текста); синтаксический анализ (построение дерева зависимостей предложения); семантический анализ (построение семантического графа текста).

Необходимой предварительной процедурой для дальнейшего анализа текста в любой системе анализа естественных языков является выделение предложений из сплошного текста. В первом приближении предложение есть последовательность символов, заканчивающаяся на символы «.», «!» или «?», однако на практике следует учитывать возможность использования точки в качестве символа сокращения и другие нюансы [4]. В разработанном графематическом алгоритме используются предопределенные наборы общепринятых («г.», «гг.», «и т. д.») и распространенных («т. к.», «т. е.», «т. н.») сокращений, а также учитывается возможность сокращения инициалов в именах собственных («А. С. Пушкин» и т. п.). В результате работы алгоритма получается массив предложений (фраз), которые в дальнейшем могут обрабатываться алгоритмами морфологического и синтаксического анализа.

Для получения тестовых заданий различных видов в настоящем подходе применяются различные виды алгоритмов синтеза вопросов. Эти алгоритмы отличаются глубиной анализа естественного языка, и их можно разделить на две группы.

Алгоритмы первой группы осуществляют замену искомого слова в предложении на комбинацию символов «?» (по каждому предложению исходного текста может быть построено не более одного вопроса). К данной группе относятся следующие алгоритмы:

- поиск сокращений (аббревиатур);
- поиск численных значений;
- генерация на основе определений;
- генерация из конструкций «если …, то …».

Алгоритмы этой группы основываются на тривиальном просмотре фразы и поиске необходимых символов (или же конкретных слов). Эти алгоритмы наиболее просто реализуются на практике и отличаются относительно высоким быстродействием, однако часто возвращают неприемлемые результаты.

Алгоритмы второй группы выполняют построение вопроса по результатам синтаксического анализа текста. Сюда относятся следующие алгоритмы:

- вопросы к подлежащему (*что?*, *кто?*);
- вопросы к прилагательным (*какой?* и т. п.);
- вопросы к обстоятельству места (*где?*);
- вопросы к обстоятельству времени (*когда?*).

Эти алгоритмы требуют наличия развитого морфологического словаря. В используемом для исследований программном продукте они реализованы с помощью библиотек *RML* (http://www.aot.ru/).

Для примера опишем алгоритм формирования вопроса к прилагательному. Блок-схема этого алгоритма представлена на рис. 1. В начале работы инициализируются переменные *A*, *B*, *C*: *A* – обрабатываемое предложение из массива текста; *B* – ответ на вопрос (присваивается пустая строка); *C* – готовое предложение для тестирования (по умолчанию равно *A*). Затем с помощью метода *FindSituation()* библиотек *RML* производится синтаксический анализ предложения *A*. Далее инициализируются вспомогательные переменные для работы алгоритма: *flag* = *истина* (сигнализатор найденной фразы для генерации вопроса), *j* = 0 (переменная цикла, номер текущего узла предложения). В переменную *K* записывается число узлов в анализируемом предложении. Затем начинается цикл с предусловием: пока *j* меньше либо равно (*K*−1) и *flag* = *истина*. В цикле последовательно разбирается каждый синтаксический узел предложения. Если в узле находится тип отношения «свойство», то с помощью метода *GetGramInfo()* находятся грамматические характеристики зависимого слова, и, если это слово представляет собой прилагательное, то оно записывается в переменную *B*. Наконец, формируется вопрос с учетом формы прилагательного (его характеристик, получаемых от указанного выше метода). Для закрытия цикла переменной *flag* присваивается значение *ложь*.

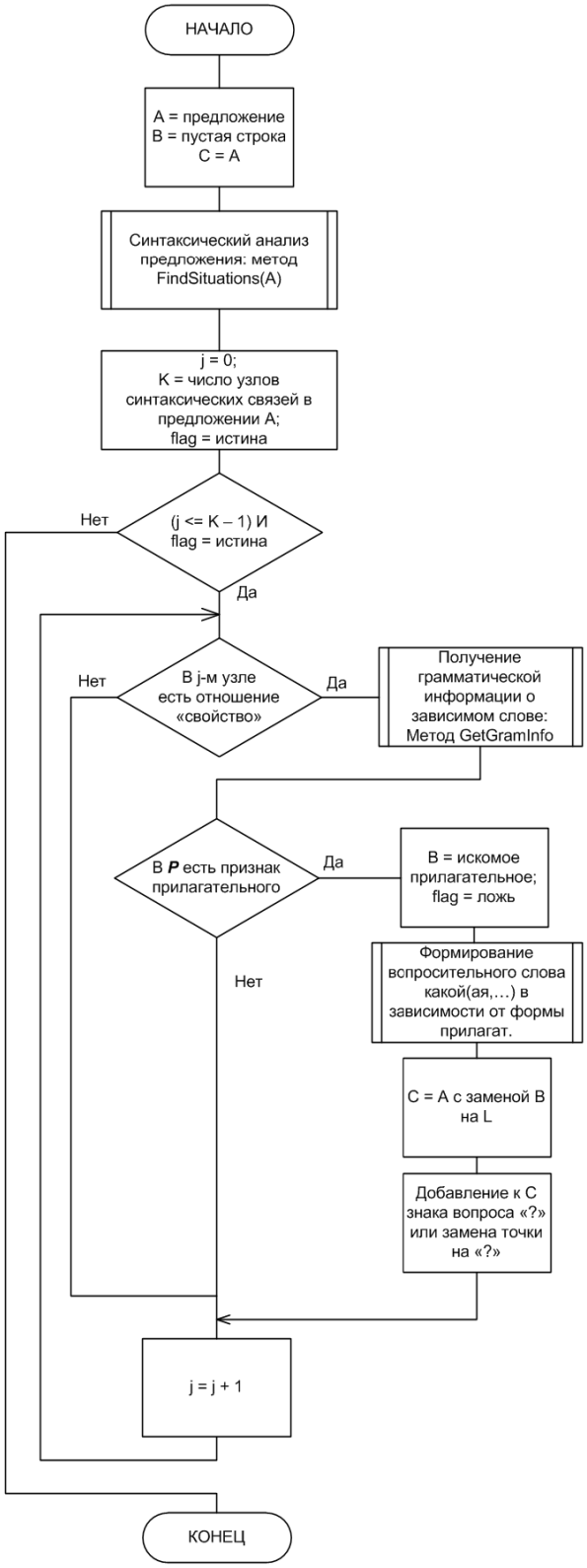

Рис. 1. Блок-схема алгоритма генерации вопросов к прилагательным

2. ЭКСПЕРИМЕНТАЛЬНАЯ АПРОБАЦИЯ

Следует отметить, что если текст обрабатывается только до уровня синтаксиса без учета семантики, не все генерируемые вопросы могут являться релевантными в данной предметной области, вследствие чего на пользователя может лечь довольно трудоемкая задача отбора вопросов. Тем не менее, испытания подхода на конкретных учебных дисциплинах показали, что даже при такой реализации подход дает относительно неплохие результаты. Например, в случае учебного пособия по дисциплине «Интеллектуальные информационные системы» (авт. Швецов А. Н.) с помощью алгоритмов второй группы удалось получить 40 % заданий, пригодных для использования в тесте без изменения, и 22,8 % заданий, из которых можно получить пригодные задания путем редактирования (т. е. больше половины полученных заданий оказались подходящими для составления тестов при той или иной степени участия преподавателя). Алгоритмы первой группы демонстрировали высокую полезность при анализе пособий, содержащих

большое количество чисел, но в иных случаях часто оказывались малоэффективными (около 15 % пригодных заданий).

Для оценки временной эффективности алгоритмов они были испытаны на компьютерах различной аппаратной конфигурации (от нетбука до мощной рабочей станции на базе процессора *Intel Core i3*). Между алгоритмами первой и второй группы наблюдалась существенная разница: вследствие более высокой сложности алгоритмов второй группы для их работы часто требуется больше времени даже в случае меньшего размера пособия. Но в целом, на всех конфигурациях генерация заданий заняла относительно мало времени: не более 2 мин. на загрузку пособия и не более 40 с на генерацию заданий тем или иным алгоритмом.

В целом, по результатам испытаний был сделан вывод, что эффективность генерации заданий будет изменяться от пособия к пособию и для достижения лучших результатов необходимо усовершенствовать подход таким образом, чтобы учитывалась семантика выделяемых из текста предложений [5].

3. ДАЛЬНЕЙШЕЕ РАЗВИТИЕ ПРОГРАММЫ: СЕМАНТИЧЕСКИЙ ПОДХОД

Для перехода на уровень семантики в настоящем подходе предлагается формировать базу знаний на основе семантических сетей, которая позволяла бы определять принадлежность текстов к той или иной предметной области и соответственно выбирать тексты для генерации тестов по той или иной учебной дисциплине. По мнению авторов, большой потенциал при реализации в современных интеллектуальных системах имеет логический подход к семантическому анализу естественных языков, который интенсивно исследуется в работах лингвистов и логиков.

При создании моделей и методов семантического анализа могут использоваться различные логические формализмы: семантика смысла и денотата Г. Фреге; теория объектов и пропозиций Б. Рассела; теория истины А. Тарского; семантика возможных миров С. Крипке; теория типов Б. Рассела и К. Айдукевича. Новое направление в этой области, получившее название формальной семантики, сформировали работы Р. Монтегю. Основная идея данного направления выражена в названии одного из его основополагающих трудов: *English as a formal language* [6]. Любой естественный язык предлагается понимать как формальный логический язык, который является более сложным по отношению к существующим формальным языкам. При описании естественного языка предлагается использовать такие же понятия и конструкции, как для других логических языков.

Логика Р. Монтегю является основой метода семантической обработки информации, который предлагается использовать в настоящем подходе [7].

Процесс обработки текстовой информации по данному методу представляется следующим образом. Входные данные для системы – это текст на естественном языке. На основе лингвистической обработки текста строится набор категорий интенсиональной логики для дальнейшего применения правил трансформации (ПТ) синтаксических конструкций в элементы единой формулы, отражающей смысл высказывания. Формальное представление не зависит от конкретного естественного языка и представляет собой набор типов и операций над формулами. Результатом обработки является формализованное представление смысла текста в виде набора формул, отражающих смысл предложений и множества постулатов значений, представляющих фоновые знания о мире. Метод заключается в применении алгоритмов формализации смысла естественно-языковых текстов, заполнении базы знаний и интерпретации на ней запросов, формируемых при генерации тестовых заданий.

В методе выделяются следующие этапы:

1) формализация естественно-языковой фразы;
2) интерпретация формулы формальной семантики;
3) заполнение базы знаний.

При формализации естественно-языковой фразы сначала проводится лингвистическая обработка фразы языковым процессором с целью построения синтаксического дерева. Далее осуществляется рекуррентный обход узлов дерева с целью применения функции генерации формулы на основе ПТ. Каждому узлу дерева сопоставляется некоторая категория интенсиональной логики. Для определения категории узла используется табличная функция отображения категорий синтаксического анализатора в категории интенсиональной логики.

Под интерпретацией формулы в формальной семантике понимается установление ее истинностного значения на основе представленной теоретико-множественной картины мира. Для корректного применения функции интерпретации в технической системе представляется необходимым разделение данных о состоянии мира и процедур интерпретации фрагментов языка логики.

Для реализации механизма заполнения базы знаний предлагается расширить толкование термина «интерпретация», используемого в формальной семантике. Под интерпретацией здесь понимается не просто вычисление истинностного значения формулы, но и отображение знакового представления выражения на определенную картину мира. При этом может проводиться как интерпретация с целью вычисления выражения, так

и интерпретация для изменения модели предметной области, на которой производится отображение. Для заполнения базы знаний на основе данных текста строится формальное представление естественно-языкового текста в виде множества набора формул интенсиональной логики. Затем определяется целевая семантическая сеть и осуществляется ее клонирование, результатом которого является семантическая сеть, имеющая пометы на всех узлах, показывающие ее принадлежность к определенному тексту. Далее происходит интерпретация каждой формулы с целью нанесения признаков объектов, представленных в формуле, на семантическую сеть.

ЗАКЛЮЧЕНИЕ

Представленная программа может быть полезна любому преподавателю, использующему для оценки знаний учащихся метод тестирования. Особенно хорошо программа подходит для случая инженерно-технических дисциплин, в которых применяются учебные пособия, отличающиеся высокой степенью структурированности материала.

Оригинальный подход к задаче генерации компьютерных тестов, применяемый в данной программе, позволит существенно облегчить работу преподавателя при составлении учебных и контрольных тестов, а также сэкономить время на проработку учебных пособий при формировании тестовых заданий.

В число основных преимуществ программного продукта входит ориентированность на совместное применение с *LMS*-системой *Moodle*. Эта особенность дает возможность использовать разрабатываемые материалы для тестирования в удаленном режиме вне зависимости от операционной системы, под управлением которой работает компьютер обучаемого (единственное условие – наличие подключения к Интернету или локальной сети).

Описанный вариант развития применяемого подхода позволяет перевести программу на уровень семантики. Усовершенствованный подход позволит автоматически извлекать новые знания из естественно-языковых текстов и формировать многоуровневые базы знаний по взаимосвязанным предметным областям. Это обеспечит эффективное накопление знаний, используемых для составления учебных тестов, а также откроет дополнительные возможности управления знаниями. В частности, в Вологодском государственном техническом университете разрабатывается проект интеллектуального агентно-ориентированного учебного комплекса, важной частью которого является база знаний, используемая не только для хранения и накопления учебных материалов, но и для обеспечения работы агентов, формирующих индивидуальные траектории обучения.

Важно отметить, что рассмотренная программа может применяться не только в учебных заведениях, но и на предприятиях, осуществляющих корпоративное обучение сотрудников.

СПИСОК ЛИТЕРАТУРЫ

1. Башмаков А. И., Башмаков И. А. Разработка компьютерных учебников и обучающих систем. – М.: Информационно-издательский дом «Филинъ», 2003. – 616 с.

2. Воронец И. В., Швецов А. Н., Алешин В. С. Универсальная автоматизированная система тестирования знаний и самообучения, основанная на анализе естественно-языковых текстов учебных пособий. – Пилотируемые полеты в космос. Сб. докл. Пятой международ. научн.-практ. конф. 9–10 апреля 2003 г. – Звездный городок Моск. обл.: РГНИИЦПК, 2003. – С. 65–67.

3. Швецов А. Н., Алешин В. С. Построение приближенной концептуальной модели предметной области на основе анализа смысла естественно-языковых текстов. – Международная конференция по мягким вычислениям и измерениям SCM’2003. Сб. докладов. Т. 2. – СПб.: Изд-во СПбГЭТУ «ЛЭТИ», 2003. – С. 120–123.

4. Riley, Michael D. Some applications of tree-based modeling to speech and language indexing. – Proceedings of the DARPA Speech and Natural Language Workshop. – Stroudsburg, PA, USA, 1989. – P. 339–352.

5. Методология создания агентно-ориентированных учебных комплексов для подготовки специалистов технического профиля: отчет о НИР (промежуточ.) / Вологодский государственный технический университет; рук. **Швецов А. Н.** ; исполн.: Горбунов В. А. [и др.]. – М., 2011. – 175 с. – Библиогр.: с. 170–175. – № ГР 01201056386.

6. Montague, R. English as a formal language / R. Montague, edited by R. H. Thomason. – Formal Philosophy. – Yale University Press, 1974.

7. Летовальцев В. И., Швецов А. Н. Программная формализация естественного языка средствами формальной семантики. – Программные продукты и системы. – 2010. – №3. – С. 85–90.