

Объединение автоматизированных учебных пособий на основе семантической сети понятий

© Э.Н. Самохвалов, Ю.Е. Гапанюк

МГТУ им. Н.Э. Баумана, Москва, 105005, Россия

Предложен алгоритм объединения автоматизированных учебных пособий на основе семантической сети понятий. Рассмотрены элементы формализованной модели автоматизированного учебного пособия на основе семантической сети понятий, необходимые для реализации алгоритма. Представлена классификация элементов модели на элементы с частичной эквивалентностью, элементы без частичной эквивалентности, ссылочные элементы. Для каждого класса элементов предложен отдельный алгоритм объединения, позволяющий разработчикам пособий независимо вносить в них изменения и затем объединять изменения в единое представление.

Ключевые слова: *автоматизированное учебное пособие, семантическая сеть понятий, элемент учебного пособия, частичная эквивалентность, алгоритм объединения.*

Введение. Одной из наиболее известных проблем разработки автоматизированных учебных пособий является проблема комплексного представления предметной области в нескольких пособиях. Часто даже близкие по смыслу учебные пособия не формируют у обучающегося единого комплексного представления о рассматриваемой предметной области. Это, в частности, связано с тем, что авторы пособий обладают каждый своим опытом в предметной области, используют различную терминологию и рассматривают изучаемые явления с разных точек зрения и т. д.

Для преодоления данной проблемы в работе [1] предложен подход, связанный с построением нескольких автоматизированных учебных пособий на основе единой семантической сети понятий.

Следует отметить, что для формализации предметных областей традиционно используются графовые модели, в том числе и для обучения. Наиболее известным подходом в этой области является теория «концептуальных карт», предложенная профессором Джозефом Новаком. Однако система «SmartTools» [2] — автоматизированный инструмент, разработанный на основе теории Новака, — предназначена только для формализации предметных областей и не позволяет создавать автоматизированные учебные пособия.

В работе [1] семантическая сеть понятий описывает основные понятия предметной области и отношения между ними, а учебное пособие задает обучающую траекторию на семантической сети понятий.

Таким образом, несколько пособий могут формироваться на основе единой семантической сети понятий предметной области. При изучении новых пособий знания обучающегося обогащаются новыми понятиями и связями между понятиями, что позволяет постепенно формировать у обучающегося непротиворечивую картину знаний о предметной области.

В работе [1] была предложена формализованная модель автоматизированного учебного пособия на основе семантической сети понятий. Однако в ней не был рассмотрен вопрос автоматизированного объединения учебных пособий. Данный вопрос является актуальным в рамках предложенной модели, так как разработчики пособий могут независимо изменять семантическую сеть понятий и учебные пособия. Внесенные изменения необходимо объединять для формирования единой версии семантической сети понятий и учебных пособий.

Данная статья построена следующим образом. Сначала в сокращенном виде приведены элементы формализованной модели из [1], без которых невозможно описание алгоритма объединения пособий (в частности, не рассматривается модель текущих знаний обучающегося, не использующая в алгоритме объединения пособий). Далее подробно рассмотрены алгоритмы объединения пособий.

Элементы формализованной модели автоматизированного учебного пособия на основе семантической сети понятий. Семантическая сеть понятий учебного пособия

$$TG = \langle TGT, TGR \rangle, TGT = \{ tgt_i \}, TGR = \{ tgr_j \}, \quad (1)$$

где TG — семантическая сеть понятий учебного пособия; TGT — множество понятий, основных элементов предметной области, в которой создается учебное пособие; tgt_i — понятие учебного пособия; TGR — множество отношений, связей между понятиями; tgr_j — отношение между понятиями.

Понятие

$$tgt_i = \langle id, eid, tgt_{NM}, URI, Q_S, Q_E \rangle, URI = \{ uri_i \}, \quad (2)$$

где id — уникальный идентификатор элемента; eid — уникальный идентификатор, изменяемый при редактировании элемента; tgt_{NM} — наименование понятия; URI — множество ресурсов, связанных с понятием; Q_S — множество предварительных контрольных вопросов; Q_E — множество завершающих контрольных вопросов; uri_i — ресурс, связанный с понятием.

Идентификаторы id и eid определяются далее для всех элементов.
Отношение между понятиями

$$tgr_j = \langle id, eid, tgr_{NM}, tgr_S, tgr_E, URI, Q_S, Q_E \rangle, URI = \{ uri_i \}, \quad (3)$$

где tgr_{NM} — наименование отношения; tgt_{S} — понятие, из которого исходит отношение; tgt_{E} — понятие, в которое входит отношение; URI — множество ресурсов, связанных с отношением.

Множество контрольных вопросов

$$Q = \langle \text{id}, \text{eid}, Q_{\text{NM}}, \{ q_i \} \rangle, \quad (4)$$

где Q — множество контрольных вопросов; Q_{NM} — наименование множества контрольных вопросов; q_i — контрольный вопрос.

В частности, множествами контрольных вопросов являются Q_{S} и Q_{E} .

Контрольный вопрос

$$q_i = \langle \text{id}, \text{eid}, \text{qt}, a_r, t_r, \{ a_{wi}, t_{wi} \} \rangle. \quad (5)$$

Здесь qt — текст вопроса; a_r — правильный вариант ответа на контрольный вопрос; t_r — текст, подтверждающий правильность ответа a_r ; a_{wi} — неправильный вариант ответа; t_{wi} — текст, объясняющий неправильность варианта ответа a_{wi} .

Множество учебных пособий

$$\text{CRS} = \{ \text{crs}_i \}, \quad (6)$$

где CRS — множество учебных пособий; crs_i — учебное пособие,

$$\text{crs}_i = \langle \text{id}, \text{eid}, \text{crs}_{\text{NM}}, \text{LG} \rangle, \text{LG} = \{ \text{lg}_i \} \quad (7)$$

(crs_{NM} — наименование учебного пособия; LG — множество учебных целей пособия; lg_i — учебная цель).

Учебная цель определяется как упорядоченное множество понятий и отношений на семантической сети понятий учебного пособия:

$$\text{lg}_i = \langle \text{id}, \text{eid}, \text{lg}_{\text{NM}}, \{ \text{tg}_i \} \rangle, \text{tg}_i \in (\text{TGT} \cup \text{TGR}), \quad (8)$$

где lg_i — учебная цель; lg_{NM} — наименование учебной цели; tg_i — элемент учебной цели, принадлежащий объединенному множеству понятий и отношений семантической сети понятий учебного пособия.

Алгоритм объединения автоматизированных учебных пособий на основе семантической сети понятий. Основные положения.

Алгоритм предназначен для объединения двух моделей автоматизированных учебных пособий на основе семантической сети понятий. Объединяются следующие элементы модели:

- семантическая сеть понятий учебного пособия (понятия и отношения);
- множества контрольных вопросов (URI);
- учебные цели и учебные пособия.

Два элемента модели из двух различных учебных пособий могут быть:

- полностью эквивалентны. В результирующем пособии должен присутствовать только один элемент (первый). Никаких дополнительных действий над элементами не требуется;
- частично эквивалентны. В результирующем пособии должен присутствовать один элемент, являющийся результатом дополнительного объединения параметров первого и второго элементов;
- неэквивалентны. В результирующем пособии должны присутствовать оба элемента.

Для удобства объединения каждому элементу модели ставится в соответствие два уникальных идентификатора: id и eid . Идентификатор id задается при создании элемента и не меняется. Идентификатор eid ($edit\ id$) меняется при каждом редактировании элемента.

Кроме того, вводятся множество эквивалентных элементов EQ, задаваемых пользователем, и множество эквивалентных элементов EQG, создаваемое в процессе работы алгоритма:

$$EQ = \{ \langle id_1, id_2 \rangle \}, EQG = \{ \langle id_1, id_2 \rangle \}, \quad (9)$$

где id_1 — идентификатор элемента из первого пособия; id_2 — идентификатор элемента из второго пособия. Предполагается, что эти элементы одного типа, например понятия или множества вопросов.

Множество EQ задается разработчиком пособия перед работой алгоритма и позволяет для улучшения точности работы алгоритма определять эквивалентные элементы учебных пособий.

Множество EQG строится динамически в процессе работы алгоритма и используется для правильного установления эквивалентности элементов.

Алгоритм объединения рассчитан на два различных случая:

1) объединяются несвязанные пособия, которые могут содержать похожие элементы. В этом случае алгоритм ориентируется на «ручное» указание эквивалентности элементов с использованием множества EQ;

2) объединяются пособия, которые являются вариантами общего исходного пособия. В этом случае алгоритм ориентируется на «автоматическое» указание эквивалентности элементов с использованием идентификаторов id и eid . Множество EQ дополнительно используется для уточнения работы алгоритма.

Для выполнения объединения разделим элементы модели на три вида.

I. Элементы модели, для которых используется частичная эквивалентность.

В основном это — составные элементы модели, содержащие вложенные параметры:

- TGT — множество понятий учебного пособия;
- TGR — множество отношений между понятиями;
- CRS — множество учебных пособий.

В этом случае два составных элемента модели из двух различных учебных пособий:

1) полностью эквивалентны, если у них совпадают идентификаторы id и eid , т. е. это — элементы, которые были унаследованы из общего учебного пособия и не редактировались;

2) частично эквивалентны в следующих случаях:

- у элементов совпадают идентификаторы id , но не совпадают идентификаторы eid , т. е. это — элементы, которые были унаследованы из общего учебного пособия, но редактировались;

- присутствуют в множестве EQ ;

- у элементов совпадают названия (если включен соответствующий режим). Этот вариант можно считать «быстрым» вариантом задания множества EQ ;

3) Неэквивалентны, если они не являются ни эквивалентными, ни частично эквивалентными.

II. Элементы модели, для которых не используется частичная эквивалентность.

В основном это — простые элементы модели, которые не имеет смысла или которые нельзя объединять частично:

- URI — множество ресурсов, связанных с понятием или отношением;

- Q — множество контрольных вопросов;

- LG — упорядоченное множество учебных целей пособия.

Для учебной цели не используется частичная эквивалентность, так как разработчик пособия определяет ее «целостно» в виде упорядоченного набора элементов и было бы неправильно произвольно заменять элементы этого набора. В этом случае два элемента модели из двух различных учебных пособий:

1) эквивалентны, если:

- у элементов совпадают идентификаторы id и eid , т. е. это — элементы, которые были унаследованы из общего учебного пособия и не редактировались;

- присутствуют в множестве EQ ;

- у элементов совпадают названия (при включении соответствующего режима). Этот вариант можно считать «быстрым» вариантом задания множества EQ .

2) неэквивалентны в остальных случаях, в том числе элементы не являются эквивалентными, если у них совпадают идентификаторы id , но не совпадают идентификаторы eid . В этом случае оба элемента добавляются в результат, и разработчик пособия в дальнейшем может редактировать их вручную.

III. Ссылочные элементы.

Это — ссылка на понятие или отношение между понятиями:

- $tgts$ — понятие, из которого исходит отношение, и tgt_E — понятие, в которое входит отношение для отношений между понятиями;

- tg_i — элемент учебной цели, принадлежащий объединенному множеству понятий и отношений между понятиями.

Для таких элементов необходимо решать проблему «сломанных ссылок».

Эта проблема может возникнуть в случае частично эквивалентных элементов для элементов вида I и полностью эквивалентных — для элементов вида II, поскольку в таких случаях из двух элементов с разными id получается один элемент, id которого в соответствии с алгоритмом равен id первого элемента. Следовательно, возможна ситуация, когда элемент из второго пособия не будет включен в результат, так как он объединен с элементом из первого пособия. При этом в соответствии с алгоритмом такие частично объединенные элементы были записаны в множество EQG. Поэтому, если какой-то параметр является ссылкой на элемент во втором пособии, необходимо проверить, не встречается ли он в множестве EQG, и если встречается, то заменить его на соответствующий элемент из множества EQG. Таким образом, все элементы из второго пособия, которые не были включены в результирующее пособие, будут заменены на соответствующие элементы результирующего пособия.

Далее детально будут рассмотрены алгоритмы объединения:

- для элементов с частичной эквивалентностью;
- для элементов модели без частичной эквивалентности;
- для ссылочных элементов;
- итоговый алгоритм объединения.

Описание алгоритма (рис.1).

Блок 1. Входными параметрами алгоритма являются: множества элементов для объединения из первого и второго пособия, логическая переменная $флаг_назв$ (определяет, нужно ли считать эквивалентными элементы с одинаковыми названиями), множества эквивалентных элементов EQ и EQG.

Блок 2. По умолчанию в результат включаются элементы из первого пособия.

Блок 3. В цикле осуществляется перебор всех элементов, принадлежащих множеству «Пособие2.Множество». Цикл завершается в блоке 14.

Блок 4. Поиск элемента «элемент1» в первом пособии, для которого $элемент.id = элемент1.id$ и $элемент.eid = элемент1.eid$, т. е. «элемент1» полностью эквивалентен текущему элементу.

Если такой элемент найден, то текущий элемент не нужно включать в результирующее пособие, так как туда уже включен полностью эквивалентный «элемент1». Переход к следующей итерации цикла: к блоку 14.

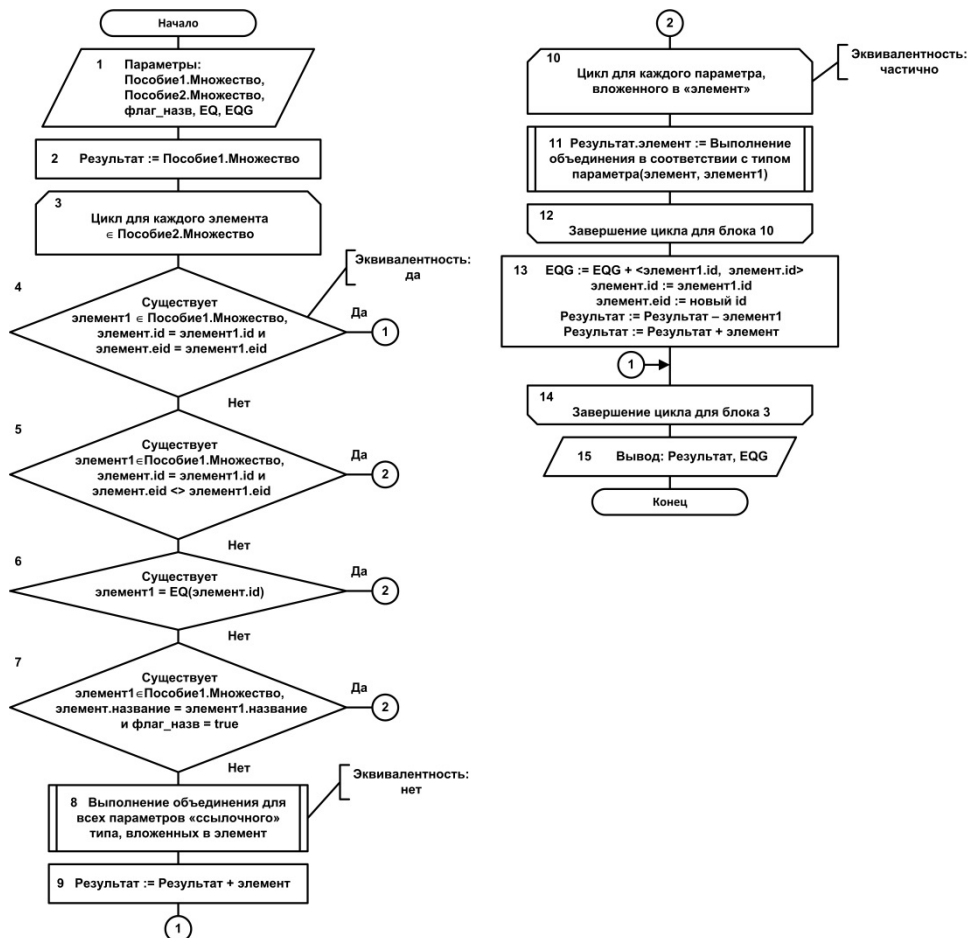


Рис. 1. Блок-схема алгоритма объединения элементов с частичной эквивалентностью

Блок 5. Поиск элемента «элемент1» в первом пособии, для которого $\text{элемент.id} = \text{элемент1.id}$ и $\text{элемент.eid} \neq \text{элемент1.eid}$, т. е. «элемент1» частично эквивалентен текущему элементу.

Если такой элемент найден, необходимо дополнительно объединять частично эквивалентные элементы. Переход к блоку 10.

Блок 6. Поиск элемента «элемент1» в множестве EQ, т. е. эквивалентного элемента, который задал пользователь.

Блок 7. Поиск элемента «элемент1» в первом пособии с таким же названием. Такой поиск проводится, только если включен поиск по названиям $\text{флаг_назв} = \text{true}$.

Если такой элемент найден, надо дополнительно объединять частично эквивалентные элементы. Переход к блоку 10.

Если ни одно из предыдущих условий не отработало, элементы неэквивалентны. Переход к блоку 8.

Блок 8. Выполнение объединения для всех параметров «ссылочного» типа, вложенных в элемент. Это необходимо, потому что элемент (соответствующий вложенному параметру) из второго пособия может быть не включен в результат, поскольку он объединен с элементом из первого пособия и возникнет «сломанная» ссылка.

Блок 9. Так как элемент и «элемент1» неэквивалентны, текущий элемент из второго пособия добавляется в результирующее множество.

Блоки 10–12. В цикле осуществляется перебор всех параметров, вложенных в текущий элемент. Для каждого параметра вызывается соответствующий алгоритм объединения.

Блок 13. В этом блоке проводится замена элемента из первого множества обновленным элементом.

Информация о замене «элемент1» на «элемент» сохраняется в множестве EQG, что дальше используется при замене элементов ссылочного типа.

Блок 14. Текущему элементу присваивается id элемента из первого множества и создается новое значение идентификатора eid. Из результирующего множества удаляется «элемент1» и добавляется обновленный элемент.

Блок 15. Выходными параметрами алгоритма являются результирующее множество «Результат» и обновленное множество EQG, содержащее информацию о замене элементов 3. Алгоритм объединения элементов без частичной эквивалентности

Описание алгоритма (Рис. 2).

Блок 1. Входными параметрами алгоритма являются: множества элементов для объединения из первого и второго пособия, логическая переменная флаг_наз (определяет, нужно ли считать эквивалентными элементы с одинаковыми названиями), множества эквивалентных элементов EQ и EQG.

Блок 2. По умолчанию в результат включаются элементы из первого пособия.

Блок 3. В цикле осуществляется перебор всех элементов, принадлежащих множеству «Пособие2.Множество». Цикл завершается в блоке 10.

Блок 4. Поиск элемента «элемент1» в первом пособии, для которого элемент.id = элемент1.id и элемент.eid = элемент1.eid, т. е. «элемент1» полностью эквивалентен текущему элементу.

Если такой элемент найден, текущий элемент не нужно включать в результирующее пособие, так как туда уже включен полностью эквивалентный «элемент1». Переход к следующей итерации цикла: к блоку 10.

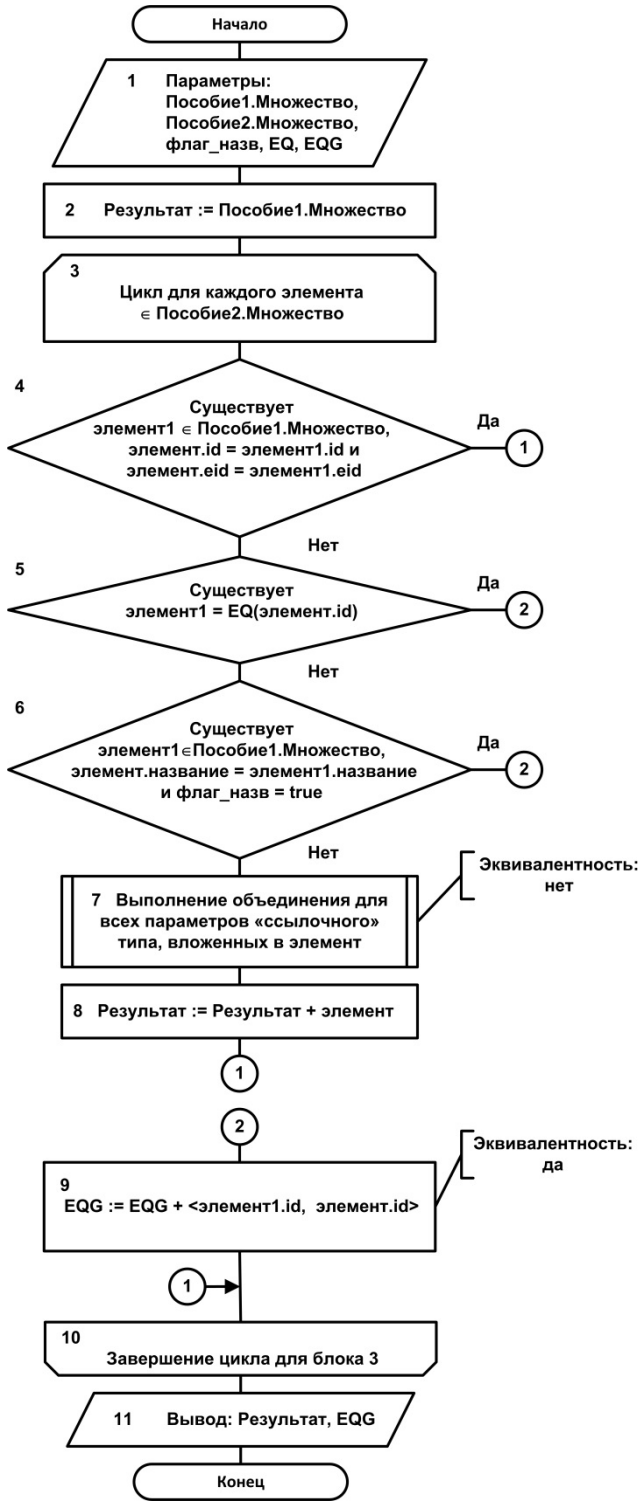


Рис. 2. Блок-схема алгоритма объединения элементов без частичной эквивалентности

Блок 5. Поиск элемента «элемент1» в множестве EQ, т. е. эквивалентного элемента, который задал пользователь.

Если такой элемент найден, то необходимо сохранить информацию о замене в множестве EQG. Переход к блоку 9.

Блок 6. Поиск элемента «элемент1» в первом пособии с таким же названием. Этот поиск проводится, только если включен поиск по названиям `флаг_назв = true`.

Если такой элемент найден, надо сохранить информацию о замене в множестве EQG. Переход к блоку 9.

Если ни одно из предыдущих условий не отработано, элементы неэквивалентны. Переход к блоку 7.

Блок 7. Объединение всех параметров «ссылочного» типа, вложенных в элемент. Это необходимо, потому что элемент (соответствующий вложенному параметру) из второго пособия может быть не включен в результат, так как он объединен с элементом из первого пособия и возникнет «сломанная» ссылка.

Блок 8. Так как элемент и «элемент1» не эквивалентны, текущий элемент из второго пособия добавляется в результирующее множество.

Блок 9. Информация о замене «элемент1» на «элемент» сохраняется в множестве EQG, что дальше используется при замене элементов ссылочного типа.

Блок 11. Выходными параметрами алгоритма являются результирующее множество «Результат» и обновленное множество EQG, содержащее информацию о замене элементов.

Описание алгоритма.

Блок 1. Входными параметрами алгоритма являются: «Множество» элементов для объединения (замены) из второго пособия, множество эквивалентных элементов EQG.

Блок 2. Результат является пустым множеством.

Блок 3. В цикле осуществляется перебор всех элементов, принадлежащих множеству. Цикл завершается в блоке 7.

Блок 4. Поиск элемента «элемент1» в множестве EQG, т. е. эквивалентного элемента из результирующего пособия, который соответствует текущему элементу.

Если «элемент1» найден, это означает, что текущий элемент не был включен в результирующее пособие, а в результирующем пособии ему соответствует «элемент1». Поэтому в блоке 5 в результирующее множество добавляется «элемент1». В противном случае в блоке 6 в результирующее множество добавляется текущий элемент.

Блок 8. Выходным параметром алгоритма является результирующее множество «Результат».

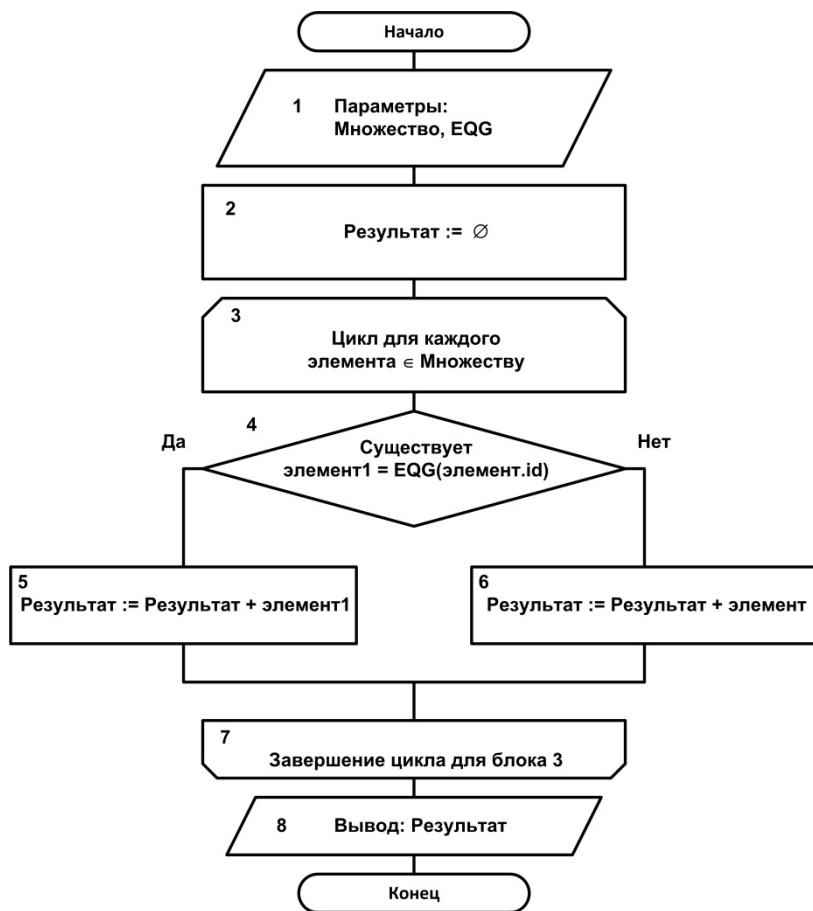


Рис. 3. Блок-схема алгоритма объединения ссылочных элементов

Описание алгоритма (Рис. 4).

Блок 1. Входными параметрами алгоритма являются: первое и второе пособия, логическая переменная флаг_наз (определяет, нужно ли считать эквивалентными элементы с одинаковыми названиями), множество эквивалентных элементов EQ, которое задал пользователь перед объединением пособий.

Блок 2. Инициализация переменных. Результирующее пособие пустое. Множество EQG пустое.

Блок 3. Для множества понятий вызывается алгоритм объединения элементов с частичной эквивалентностью.

В качестве входных параметров ему передаются множества понятий для первого и второго пособий, множество EQ, заполненное пользователем, и множество EQG. Множество EQG также является выходным параметром, так как в процессе работы алгоритма в нем сохраняются все замены элементов второго пособия на элементы первого пособия. Кроме того, выходным параметром является множество понятий результирующего пособия.

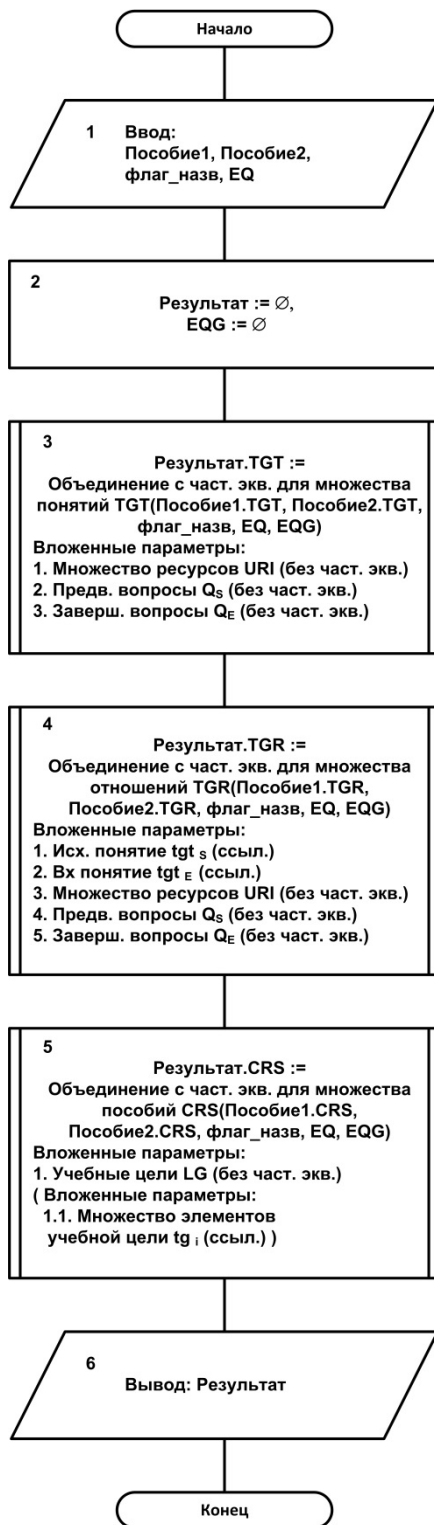


Рис. 4. Блок-схема итогового алгоритма объединения автоматизированных учебных пособий

Вложенными параметрами для понятия учебного пособия являются:

1) множество ресурсов URI. Для этого параметра вызывается алгоритм объединения элементов без частичной эквивалентности;

2) множество предварительных контрольных вопросов Q_S . Для этого параметра вызывается алгоритм объединения элементов без частичной эквивалентности;

3) множество завершающих контрольных вопросов Q_E . Для этого параметра вызывается алгоритм объединения элементов без частичной эквивалентности.

Блок 4. Для множества отношений вызывается алгоритм объединения элементов с частичной эквивалентностью.

В качестве входных параметров ему передаются множества отношений для первого и второго пособий, множество EQ (заполненное пользователем) и множество EQG. Множество EQG также представляет собой выходной параметр, поскольку в процессе работы алгоритма в нем сохраняются все замены элементов второго пособия на элементы первого пособия. Кроме того, выходным параметром является множество отношений результирующего пособия.

К вложенным параметрам для отношения учебного пособия относятся:

1) понятие, из которого исходит отношение tgt_S . Для этого параметра вызывается алгоритм объединения ссылочных элементов;

2) понятие, в которое входит отношение tgt_E . Для этого параметра вызывается алгоритм объединения ссылочных элементов;

3) множество ресурсов URI. Для этого параметра вызывается алгоритм объединения элементов без частичной эквивалентности;

4) множество предварительных контрольных вопросов Q_S . Для этого параметра вызывается алгоритм объединения элементов без частичной эквивалентности;

5) множество завершающих контрольных вопросов Q_E . Для этого параметра вызывается алгоритм объединения элементов без частичной эквивалентности.

Блок 5. Для множества пособий вызывается алгоритм объединения элементов с частичной эквивалентностью.

В качестве входных параметров ему передаются множества пособий для первой и второй моделей пособия, множество EQ, заполненное пользователем, и множество EQG. Множество EQG также представляет собой выходной параметр, поскольку в процессе работы алгоритма в нем сохраняются все замены элементов второго пособия на элементы первого пособия. Кроме того, выходным параметром является результирующее множество пособий.

В этом случае вложенный параметр — множество учебных целей LG. Для этого параметра вызывается алгоритм объединения элементов без частичной эквивалентности.

Для каждой учебной цели в качестве вложенного параметра выступает множество элементов учебной цели tg_i . Для этого параметра вызывается алгоритм объединения ссылочных элементов.

Блок 6. Выходным параметром алгоритма является результирующее пособие «Результат».

Алгоритм может допускать «неточности» при объединении, принимая эквивалентные элементы за неэквивалентные и наоборот. Поэтому в нем предусмотрена возможность для разработчика пособия явно задавать эквивалентные элементы с помощью множества EQ.

Выводы. Одна из наиболее известных проблем разработки автоматизированных учебных пособий заключается в том, что даже близкие по смыслу учебные пособия не формируют у обучающегося единого комплексного представления о рассматриваемой предметной области.

Для решения этой проблемы в работе [1] предложена модель автоматизированного учебного пособия на основе семантической сети понятий.

В данной статье в рамках разработанной модели представлен алгоритм объединения автоматизированных учебных пособий на основе семантической сети понятий.

Для реализации алгоритма предложено разделить элементы модели на три различных вида эквивалентности: элементы с частичной эквивалентностью, элементы без частичной эквивалентности, ссылочные элементы. Для каждой категории рассмотрен отдельный алгоритм объединения элементов.

Использование разных видов эквивалентности для различных элементов пособия позволило реализовать алгоритм объединения пособий в виде действий над множествами элементов пособия.

Предложенный алгоритм дает возможность разработчикам пособий независимо изменять семантическую сеть понятий и учебные пособия, затем объединять изменения в единое целостное представление.

ЛИТЕРАТУРА

- [1] Самохвалов Э.Н. Модель автоматизированного учебного пособия на основе семантической сети понятий. *Инженерный журнал: наука и инновации*, 2012, вып. 3. URL: <http://engjournal.ru/catalog/it/asu/107.html>
- [2] Novak J. D., Cañas A. J. The Origins of the Concept Mapping Tool and the Continuing Evolution of the Tool. *Information Visualization Journal*, 2006, vol. 5(3), pp. 175–184.

Статья поступила в редакцию 24.06.2013

Ссылку на эту статью просим оформлять следующим образом:

Самохвалов Э.Н., Гапанюк Ю.Е. Объединение автоматизированных учебных пособий на основе семантической сети понятий. *Инженерный журнал: наука и инновации*, 2013, вып. 11. URL: <http://engjournal.ru/catalog/it/hidden/1068.html>

Самохвалов Эдуард Николаевич родился в 1936 г., окончил МВТУ им. Н.Э. Баумана в 1957 г. Канд. техн. наук, профессор кафедры «Системы обработки информации и управления» МГТУ им. Н.Э. Баумана. Автор 50 научных работ в области проектирования информационных систем в сфере обучения. e-mail: eduard.samohvalov@yandex.ru

Гапанюк Юрий Евгеньевич родился в 1974 г., окончил МГТУ им. Н.Э. Баумана в 1998 г. Канд. техн. наук, доцент кафедры ИУ-5 МГТУ им. Н.Э. Баумана. Автор 15 научных работ в области проектирования автоматизированных систем. e-mail: garyu@bmstu.ru