Обробка інформації в складних організаційних системах

УДК 004.91 /.822/.045

В.Б. Деева, Т.В. Дуравкина, А.Г. Морозова

Харьковский национальный университет имени В.Н. Каразина, Харьков

МЕТОД ГЕНЕРАЦИИ МОДЕЛИ ПРЕДМЕТНОЙ ОБЛАСТИ ПО ТЕХНИЧЕСКОЙ ДОКУМЕНТАЦИИ

В работе предложен метод автоматической генерации модели предметной области по технической документации на английском языке. Для генерации модели предметной области в виде диаграммы классов было предложено использование объектно-ориентированного подхода совместно с NLP-ориентированными инструментами. В рамках этого подхода разработан метод построения модели предметной области, основанный на принципе объектно-ориентированного анализа, и усовершенствованный путем применения онтологии предметной области и лингвистических шаблонов.

Ключевые слова: *онтология, моделирование, модель предметной области, объектно-ориентированный анализ, NLP инструменты, диаграмма классов.*

Введение

Независимо от выбранной стратегии разработки информационной системы [1 – 3] первым этапом является этап анализа предметной области и формирование требований к будущей системе. Результатом первого этапа является концептуальная модель предметной области, которая включает: функциональную модель, модель состояний и информационную (структурно-логическую) модель системы.

Существуют два основных подхода к процессу построения концептуальной модели предметной области, отличающихся критериями декомпозиции: функционально-модульный (структурный) и объектно-ориентированный [4].

Функционально-модульный [5] подход основан на принципе алгоритмической декомпозиции с выделением функциональных элементов и установлением строгого порядка выполняемых действий.

Объектно-ориентированный [6] подход основан на объектной декомпозиции с описанием поведения системы в терминах взаимодействия объектов.

Недостатком структурного подхода является невозможность повторного использования, в связи с чем, в настоящее время наибольшее распространение получил именно объектно-ориентированный подход.

При объектно-ориентированном подходе к анализу предметной области выделяются объекты, обладающие определенными свойствами и вступающие во взаимодействие между собой. Поиск «правильных» категорий объектов требует больших усилий, при этом само понятие «правильности» не удается четко сформулировать. Очень непросто выработать устойчивую систему абстракций, определяющую соответствующие объекты, описать их взаимодействие. При этом надо предвидеть сложности реализации объектов, обеспечивающей их повторное использование при проектировании других систем. Именно повторное использование и простота модификации гарантирует эффективность объектно-ориентированного программирования.

Основные сложности в объектно-ориентированном анализе проявляются на стадии анализа технической документации, написанной на естественном языке. Во-первых, естественный язык является вольным по своей природе; постоянно возникают новые слова, термины, формы слов. Во-вторых, шаблоны предложений естественного языка могут быть сложными и неоднозначными, что может привести к множеству трактовок. В-третьих, основанные на неоднозначности и сложности описаний требований на естественном языке, неявные требования могут быть сложными для понимания. И последней проблемой является то, что такие описания могут содержать большой объем информации, анализ которой занимает много времени.

Ранние исследование продемонстрировали способность NLP-ориентированных инструментов [7] поддерживать выявление неоднозначности и обеспечивать многоразовое использование знаний. В последних проектах исследователи сосредоточили свое внимание на извлечении объектно-ориентированных концептов для генерации статических и динамических представлений системы из пред-

метно-ориентированных описаний на естественном языке.

Таким образом, использование объектно-ориентированного подхода совместно с NLP-ориентированными инструментами позволит автоматизировать процесс построения модели предметной области.

Целью настоящей статьи является построение метода автоматической генерации модели предметной области по технической документации на английском языке.

Обзор методов построения моделей предметной области

Различные методики объектно-ориентированного анализа имеют общие черты, но отличаются акцентами на определенные аспекты моделирования предметной области.

Обычно процесс построения модели предметной области разделяется на пять этапов:

1. Идентификация кандидатов на роль классов. Набор кандидатов на роль классов является источником для определения действительно необходимых классов.

2. Уточнение кандидатов на роль классов. На данном этапе происходит уточнение или исключение ненужных классов кандидатов.

3. Выявление атрибутов. Атрибуты используются для описания классов в предметной области и хранения важной информации классов.

4. Идентификация связей. Через ассоциации между классами осуществляется их взаимодействие друг с другом, для выполнения определенной функциональности системы.

5. Идентификация наследования. Наследование представляет общие и специальные черты между несколькими классами. Механизм наследования привносит большое количество преимуществ в объектно-ориентированное программирование.

Выделяют четыре основных метода объектно-ориентированного анализа [8]:

1. Метод Коада-Йордана.
2. Семишаговый метод КРБ.
3. Метод Айвара Якобсона.
4. Метод Крэга Лармана.

Алгоритм каждого из этих методов базируется на пяти этапах процесса объектно-ориентированного анализа, описанных выше.

Среди NLP-ориентированных инструментов и методов построения модели предметной области можно выделить следующие:

1. NLP-система **LOLITA** для генерации объектной модели. Такой подход определяет каждое существительное как объект, а техника семантических связей используется для выявления отношений между обнаруженными объектами. LOLITA построена на крупной семантической сети. Минусом этой системы является отсутствие различий между классами, атрибутами или объектами в семантической сети. Таким образом, данный подход ограничен и не может использоваться для выявления классов.

2. Метод, основанный на лингвистических различиях между Subject, Object и Verb, используя синтаксическую S-V-O-структуру предложения в английском языке.

3. NLP-ориентированный CASE-инструмент **CM-Builder** [9], который создает исходную модель классов из текста на английском языке. Данный инструмент хоть и выделяет классы, а не объекты, но отсутствует этап уточнения классов-кандидатов. Данный этап должен быть проделан вручную. Для выделения связей и атрибутов используются лишь эвристические правила, которые не могут предоставить полный список атрибутов и связей.

4. Инструмент **RECORD** автоматически конструирует объектную модель, основываясь на заранее определенных ключевых словах в описаниях вариантов использования. Глаголы в ключевых словах описывают поведение, а существительные трансформируются в объекты.

5. Система **LIDA** позволяет построить UML диаграмму классов на основании описаний на английском языке. Как и в предыдущем подходе, объектно-ориентированные концепты не выделяются автоматически и требуют значительного участия пользователя для их выделения.

6. **GOOAL** – прототип инструмента, разработанного Перезом-Гансалезом и Калитой, поддерживает автоматическое объектно-ориентированное моделирование из описания постановки задачи на английском языке в нотации UML, а также конструирует оба представления: статическое и динамическое. Основой методологии являются thete roles (тета-роли) и разработанный ими полуформальный язык 4WL.

Недостатками рассмотренных инструментов являются:

1. Отсутствие этапа уточнения кандидатов на роль классов.

2. Онтологии предметной области, которые несут в себе важную информацию о структуре и функционировании системы, не используются ни на одном их этапов.

3. Выделение связей между концептами происходит вручную, либо некорректно, основываясь лишь на структуре предложения.

4. Нет этапа выделения семантических пар концептов, что снижает вероятность выделения только релевантных классов.

В статье [10] был предложен подход, который опирается на комбинирование методов построения

моделей предметных областей, но соответствующая разработка не получила дальнейшего развития.

В связи с этим возникла идея разработки усовершенствованного метода построения моделей предметной области, который бы опирался на приведенный набор этапов, а также использовал достоинства каждого из уже существующих методов на каждом из этапов.

Усовершенствованный метод построения модели предметной области

Основой предлагаемого метода является комбинирование описанных ранее методов построения модели предметной области. NLP-техники это: синтаксический, семантический, морфологический и лексический анализы текста. Они взаимодействуют с 5 блоками, которые представляют собой 5 этапов формирования модели предметной области.

На вход методу автоматической генерации модели предметной области по технической документации на английском языке подается спецификация с требованиями и структурированная онтология. В результате работы метода генерируются следующие компоненты диаграммы классов: классы, атрибуты каждого класса, межклассовые связи. Межклассовые связи разделяются на: обобщение, агрегацию и ассоциацию. Ассоциация в свою очередь делится на связи «один-к-одному», «один-ко-многим» и «многие-ко-многим».

В предложенном подходе автоматической генерации модели предметной области по технической документации применяется NLP-ориентированная модель паука для поиска основных классов предметной области. Поиск же остальных классов осуществляется среди классов, связанных с основными классами. Сильной стороной данного метода является идентификация классов и связей между ними за один шаг.

На рис. 1 представлена модель паука, на которой отображены важные компоненты предложенного подхода и их взаимодействие между собой.

Ниже описан каждый компонент предложенного подхода.

Онтология предметной области служит для представления знаний о предметной области.

На выходе из блока «Выявление классов-кандидатов» получаем список предварительных кандидатов (Preliminary Candidates) и уточненных кандидатов (Refined Candidates), которые служат входными данными для модели паука. Список и уточненных кандидатов, элементы которого семантически связаны с концептами, определенными в онтологии, формируется с помощью семантических сетей и техники устранения лексической многозначности слова.

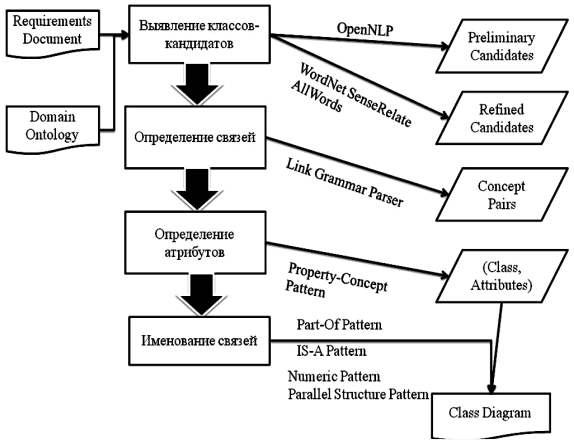

Рис. 1. Структура метода автоматической генерации модели предметной области

Блок «Определение связей» является одним из самых важных в модели паука. В данном блоке используется понятие связывающего расстояния для определения всех пар концептов, между которыми присутствует сильная семантическая связь в предложении. Для каждой определенной пары концептов назначается свой вес для обозначения степени связи между компонентами предложения.

В блоке «Определения атрибутов» происходит выделение атрибутов из классов. После того как найдены пары концептов (Concept Pairs) с сильной внутренней связью, определяется, являются ли концепты классами либо атрибутами классов.

В блоке «Именования связей» применяются лингвистические шаблоны для поиска отношений агрегации и обобщения. Используется числовая техника понимания связей для определения связей типа: один-к-одному, один-ко-многим или многие-ко-многим.

Параллельные структуры (Parallel Structure Pattern) используют особенность структуры предложения, записанного на английском языке, для выявления атрибутов, наследников класса и других частей классов.

Пошаговая детализация метода построения моделей предметной области

Метод генерации модели предметной области по технической документации на английском языке состоит в итеративном процессе выявления классов и атрибутов. На каждом шаге выбирается один концепт из определенного множества с максимальным значением веса связи, который связан с выявленными классами. Вес связи показывает, как текущий концепт семантически связан со всеми остальными концептами множества классов-кандидатов. Как только вес связи становится меньше установленного порога, алгоритм останавливается. В противном

случае концепт добавляется к множеству классов или атрибутов, в зависимости от количества его свойств.

Формирование онтологии предметной области. Онтологии предназначены для сбора знаний о предметной области. Она состоит из структурированных концептов, семантически связанных друг с другом. Предлагаемая онтология строится на основании предыдущих разработок Веларди и Фабриани (OntoLearn) [11]. В построенной онтологии присутствуют как вертикальные, так и горизонтальные связи. Вертикальные связи включают в себя: $broader(B)$, $part-of(PA)$, $instance-of(I)$. Горизонтальные: $similarity(S)$, $relatedness(R)$. Каждый концепт C в онтологии записан в следующем виде:

$$C = \{N, D, CT, B, PA, HAS\text{-}PART, I, HAS\text{-}INSTANCE, S, R\},$$

где N – название концепта;

D – описание концепта;

CT – категория концепта (основные категории, кратко описывающие концепт);

B – broader-концепт (или ссылки на более общие понятия);

PA – ряд ссылок на концепты-компоненты;

I – концепт является экземпляром;

S – набор терминов, которые представляют аналогичные концепты;

R – множество ссылок на связанные понятия.

Классификация основных категорий концептов:

Roles (Роли) – роли людей. Они представляют актеров, которые выполняют важные функции или виды деятельности.

Organizations (Организации) – представляют собой важные бизнес-единицы, такие как кампании, функциональные разделения, отделы и т.п.

Processes (Процессы) – представляют собой важные виды деятельности, которые записывают некоторую информацию в рамках возникновения события. Это могут быть транзакции, как оплата, или взаимодействие между двумя классами, как бронирование билета самолета.

Objects (Объекты) – представляют физические (материальные) вещи или нематериальные. Например, продукт, проект, счет и т.п.

Выявление классов-кандидатов. Источником классов-кандидатов являются в большинстве случаем именные фразы, реже – глагольные. Для определения части речи и разбора предложения используется OpenNLP. OpenNLP – это библиотека обработки текстов, записанных на английском языке.

В результате разбора предложения с помощью OpenNLP формируется набор именных и глагольных фраз, которые становятся предварительными кандидатами на роль классов (Preliminary Candidates). Большинство классов-кандидатов, выделенных на первом этапе, не являются корректными классами-кандидатами, более того, они могут не соответствовать предметной области. Поэтому для уточнения классов-кандидатов, используется онтология предметной области.

Для уточнения классов-кандидатов используется WordNet – электронная семантическая сеть для английского языка. WordNet представляет собой словарь, состоящий из 4 сетей для основных частей речи: существительных, глаголов, прилагательных, наречий. Базовой единицей в WordNet является не отдельное слово, а так называемый синонимический ряд или синсет, объединяющий слова со схожими значениями и по сути своей являющимися узлами сети.

Алгоритм уточнения классов-кандидатов представлен ниже.

Шаг 1: Разрешение лексической многозначности концепта в онтологии с помощью WordNet SenseRelate AllWords.

Шаг 2: Построение множества связанных концептов с целевым концептом.

Шаг 3: Определение уточненного списка классов-кандидатов.

Определение связей. Этап определения связей заключается в обнаружении любой семантически связанной в предложении пары концептов. Связи внутри пары концептов условно разделяются на три категории:

1. Связь между классами.
2. Атрибут класса.
3. Значение атрибута.

Связь между классами разделяется на обобщение, агрегацию и ассоциацию. А отношение ассоциации может быть представлено связью один-к-одному, один-ко-многим или многие-ко-многим. Тем не менее, на этом этапе не определяется конкретный тип отношения. Основная цель – определить все возможные пары концептов, семантически связанные друг с другом, и выяснить, насколько сильна связь между ними.

Так как установление семантических связей – достаточно сложная задача, даже для текста на английском языке, для измерения семантической связи двух концептов будет использоваться понятие расстояния связи или linkage distance.

Понятие Linkage Distance определено в парсере Link Grammar [12]. Link Grammar Parser – это синтаксический анализатор на основе грамматики связей. Анализатор реализован в виде программы и библиотеки на языке C.

На вход подается предложение, на выходе же получаем все связи, выделенные в предложении, а также дерево составляющих.

Определение атрибутов. После того как найден концепт, имеющий сильную семантическую связь с существующим классов, все еще необходимо определить, будет ли найденный концепт новым классом, либо атрибутом уже существующего.

Во всех рассмотренных методах построения модели предметной области одним из общих принципов выделения атрибутов является следующий принцип:

Если с найденный концепт имеет более одного свойства, то найденный концепт нужно моделировать как новый класс, в противном случае – как атрибут.

Для определения связи «свойство-концепт» используется 7-ми кортежные лингвистические шаблоны следующего вида:

{first, second, first-role, first-prep, second-role, second-prep, verb}, где

- *first* – первый концепт из пары;
- *second* – второй концепт из пары;
- *first-role* – роль первого концепта в предложении;
- *first-prep* – предлог, стоящий перед первым концептом, если такой присутствует;
- *second-role* – роль второго концепта в предложении;
- *second-prep* – предлог, стоящий перед вторым концептом;
- *verb* – глагол, который используется для формирования S-V-O структуры, если два концепта являются подлежащим и дополнением соответственно.

Шаблоны:

1. {concept, property, pre-noun modifier, null, modifier noun, null, null}

Пример: Any overdue fee is added up to *customer's outstanding balance*.

2. {property, concept, noun, null, post pp, for, null}

Пример: The *bar code ID* for each *item* is entered and video information from inventory is displayed.

3. {property, concept, object, null, pp, in, null}

Пример: System determines the rental price and due date and displays *title, due date* and *price* in the *transaction*.

4. {property, concept, object, null, pp, for, null}

Пример: A store employee creates an *account* for new *customer* by inputting customer information.

Именование связей. В стандарте UML определены следующие виды межклассовых отношений: агрегация, обобщение, ассоциация.

Для выявления связей вида агрегация и обобщение существуют строгие лингвистические шаблоны [13], которые были дополнены следующими критериями:

1. Части составного класса и наследники суперкласса должны возникать в параллельной структуре.

2. По крайней мере, один из элементов в параллельной структуре имеет связь *part-of* или *is_a* с суперклассом (определяется в процессе поиска в WordNet).

Для связи вида ассоциации применяются простейшие алгоритмы для выявления трех подтипов отношений. Для этого проверяться численные выражения или модификаторы концептов.

Параллельные структуры. Данный алгоритм в полном объеме использует данные о предметной области, представленные в виде онтологии. Тем не менее, с целью применения онтологий для нескольких задач, в нее могут быть включены лишь общие знания. Основные или важные концепты предметной области, а также их атрибуты, часто скрыты в онтологиях. Следовательно, из них можно выделить не все концепты и атрибуты, необходимые для решения определенной задачи. Однако, параллельная структура предложения, записанного на английском языке, помогает выявить пропущенные концепты и атрибуты на первых этапах анализа. Таким образом, в параллельной структуре могут быть найдены:

а) недостающие части составного класса;
б) наследники суперкласса;
в) недостающие атрибуты класса.

Например:

Дано предложение: «Rental fees can be paid by either cash, check or a major credit card.»

Если «*credit card*» уже был определен как наследник класса «*payment*» на предыдущих этапах, то, основываясь на параллельной структуре, классы «*cash*» и «*check*» также являются наследниками класса payment.

Использование параллельной структуры проходит в 2 этапа:

1. Использование лингвистических шаблонов для выявления параллельной структуры предложения, например:

Слова: *and, or, etc*.

Фразы: *both…and…, …as well as…, either…or…, not only…but also…*

2. Если один из элементов параллельной структуры имеет связь с ранее определенным концептом, тогда оставшиеся элементы также имеют связь с этим концептом.

Заключение

В работе был предложен метод автоматической генерации модели предметной области по технической документации на английском языке, который состоит из следующих этапов:

1. Формирование онтологии предметной области.

2. Выявление классов-кандидатов.
3. Уточнение классов-кандидатов.
4. Определение связей.
5. Определение атрибутов.
6. Именование связей.

Использование онтологий предметных областей и словаря улучшает поиск релевантных классов, а использование лингвистических шаблонов – поиск атрибутов и выявление связей. Путем выявления семантических пар концептов облегчается поиск связей и атрибутов.

Для каждого из этапов был предложен инструмент, реализующий необходимую функциональность каждого из этапов.

К преимуществам разработанного метода можно отнести следующее:

1. Каждый этап выполняется автоматически, без вмешательства пользователя.

2. Использование онтологии предметной области и словаря WordNet улучшает поиск релевантных классов в технической документации.

3. Использование онтологий и лингвистических шаблонов улучшает поиск связей между концептами и выделение правильного набора атрибутов.

4. Выделение семантических пар концептов облегчает поиск атрибутов, релевантных классов и установление иерархии.

Недостатки метода связаны с неразрешенными проблемами английского языка. Методы и инструменты морфологического и лексического анализа находятся еще на этапах разработки. Лингвистические шаблоны для выделения связей и атрибутов также дополняются.

Список литературы

1. Eriksson A.H. Business Modeling with UML: Business Patterns at work / A.H. Eriksson, M. Penker. – Wiley Computer Publishing, 2000.

2. Pfleeger S.L. Software Engineering: Theory and Practice (2nd ed.) / S.L. Pfleeger. – Upper Saddle River, NJ:Prentice Hall, 2001.

3. Wille R. Conceptual Graphs and Formal Concept Analysis, Conceptual Structures / R. Wille. – Berlin: Fulfilling Peirce's Dream. Proc. ICCS'97, Springer. – P. 290-303.

4. Цикритзис Д. Модели данных / Д. Цикритзис, Ф. Лоховски. – М.: Финансы и статистика, 1985. – 326 с.

5. Палмер С.Р. Практическое руководство по функционально-ориентированной разработке ПО / С.Р. Палмер, Дж.М. Фелсинг. – М.: Вильямс, 2002. – 304 с.

6. Larman C. Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development / C. Larman. – Prentice Hall PTR, Upper Saddle River, NJ, 2002.

7. Jurafsky D. Speech and Language Processing: An Introduction to Natural Language Processing, Speech Recognition, and Computational Linguistics. 2nd edition / D. Jurafsky, J. H. Martin. – Prentice-Hall, 2008.

8. Juhani Iivari. Object-orientation as structural, functional and behavioural modelling: a comparison of six methods for object-oriented analysis / Juhani Iivari. - Information & Software Technology 37(3). – P. 155-163.

9. Harmain H.M. CM-Builder – A Natural Language-based CASE Tool / H.M. Harmain. - Kluwer Academic Publishers, 2001.

10. Lopez M. How to Use Linguistic Instruments for Object-Oriented Analysis / M. Lopez, N. Juristo, A. Moreno // IEEE Software, 2000.

11. Paola Velardi, Roberto Navigli. Evaluation of OntoLearn, a methodology for automatic learning of domain ontologies / Paola Velardi, Roberto Navigli. – IOS Press, 2003.

12. Shah C. Improving Document Vectors Representation using Semantic Links and Attributes / C. Shah, P. Bhattacharyya // International Conference on Natural Language Processing (ICON). – 2003.

13. Kaindl H. How to identify binary relations for domain models / H. Kaindl. – USA:IEEE Computer Society Washington, 1996. – P. 28-36.

Поступила в редколлегию 16.10.2013

Рецензент: д-р техн. наук, доцент С.С. Танянский, Харьковский национальный университет радиоэлектроники, Харьков.

МЕТОД ГЕНЕРАЦІЇ МОДЕЛІ ПРЕДМЕТНОЇ ОБЛАСТІ ПО ТЕХНІЧНІЙ ДОКУМЕНТАЦІЇ

В.Б. Дєєва, Т.В. Дуравкіна, А.Г. Морозова

В роботі запропоновано метод автоматичної генерації моделі предметної області з технічної документації на англійській мові. Для генерації моделі предметної області у вигляді діаграми класів було запропоновано використання об'єктно-орієнтованого підходу спільно з NLP-орієнтованими інструментами. У рамках цього підходу розроблено метод побудови моделі предметної області, заснований на принципі об'єктно-орієнтованого аналізу і удосконалений шляхом застосування онтології предметної області та лінгвістичних шаблонів.

Ключові слова: *онтологія, моделювання, модель предметної області, об'єктно-орієнтований аналіз, NLP інструменти, діаграма класів.*

THE METHOD OF GENERATING A DOMAIN MODEL BASED ON THE TECHNICAL DOCUMENTATION

V.B. Deeva, T.V. Duravkina, A.G. Morozova

Automatic generation method of the domain model to the technical documentation in English is considered. Use of object-oriented approach with NLP-based tool to generate the domain model as a class diagram is proposed. The method of construction of the domain model based on the principle of object-oriented analysis and improved by the use of ontology and linguistic patterns is suggested.

Keywords: *ontology, modeling, domain model, object-oriented analysis, NLP tools, class diagram.*