OSTIS-2014
(Open Semantic Technologies for Intelligent Systems)

УДК 004.8 + 004.4

О ПОДХОДЕ К ПОДБОРУ DSL НА ОСНОВЕ АНАЛИЗА КОРПУСА СПЕЦИАЛИЗИРОВАННЫХ ДОКУМЕНТОВ

Валеев М.Т., Елохов Е.С., Узунова Е.Н., Югов А.С.

Национальный исследовательский университет «Высшая школа экономики», г. Пермь, Российская Федерация

mt.vallev.1992@gmail.com, eugene.yelokhov@gmail.com, palgonuri@gmail.com, yugovas@live.ru

На сегодняшний день многие разработки ведутся с использованием различных DSL. При этом, чтобы использовать DSL, его надо либо создать, либо взять уже существующий. Создание нового языка требует, в большинстве случаев, определенных временных и финансовых затрат, что является экономически невыгодным. Выбор подходящего варианта из уже существующих DSL является трудоемким, т.к. перебор всевозможных DSL и определение того, подходит ли конкретный DSL для решения поставленной задачи выполняется человеком «вручную». Основная причина этой проблемы в том, что не существует хранилища DSL, и не существует инструмента подбора DSL для решения конкретной задачи. В данной статье предложен подход к реализации автоматического определения требований, которым должен удовлетворять DSL (требования формируются в виде онтологии) и автоматического подбора DSL, который в наибольшей мере подходит для решения конкретной задачи.

Ключевые слова: онтология; предметно-ориентированный язык; семантическая близость слов.

Введение

В настоящее время при разработке информационных систем широко используются технологии, основанные на применении метамоделирования и предметно-ориентированных языков (DSL, Domain Specific Languages) [Лядова, 2010]. При этом, DSL создается для решения какой-либо определенной задачи. Практически о любой возникающей задаче можно сказать, что ранее решалась похожая, и, возможно, она была решена. В нашем случае это означает, что необходимый для решения поставленной задачи DSL уже был реализован, либо был реализован некий DSL, но он не полностью удовлетворяет новым условиям. Получается, что можно либо найти уже готовый DSL, либо взять разработанный ранее и внести изменения, что, конечно, требует меньших затрат, чем разработка нового DSL.

Итак, для того, чтобы выбрать один из существующих DSL, необходимо:

1. Определить требования к целевому DSL.
2. Определить, насколько каждый из имеющихся DSL соответствует сформулированным требованиям.

Требования можно определить, проанализировав документы, относящиеся к данной предметной области или к постановке и решению данной задачи. Требования предлагается представлять в виде онтологической модели, разработанной на основе анализа выбранных документов.

DSL подбирается на основе определения подобия онтологии, составленной пользователем и онтологией, построенной на основе DSL. Очевидно, для того чтобы DSL можно было сопоставить, все языки должны быть описаны в едином формате. В качестве такого формата возьмем представление, используемое в системе «MetaLanguage» [Sukhov and Lyadova, 2012].

В качестве входных данных выступают:

- Корпус документов, относящихся к определенной предметной области.
- Набор описаний DSL.

В качестве выходных данных должен быть получен список DSL, подходящих для решения поставленной задачи, упорядоченный по мере соответствия предъявленным требованиям.

В данной работе рассматриваются методы автоматизации определения требований к DSL и методы определения того, насколько каждый из DSL удовлетворяет предъявленным требованиям.

1. Существующие системы составления онтологий и трансформации моделей

На сегодняшний день имеются информационные системы, которые позволяют отдельно либо составлять онтологические модели документов на основе текстов, либо задавать соответствия онтологических моделей, тем самым преобразовывать одну модель в другую. Удалось найти два веб-ресурса, позволяющих автоматизировать разработку онтологий онтологии: OwlExporter и OntoGrid.

Основная идея OwlExporter состоит в том, чтобы проанализировать текст на естественном языке, составить краткую аннотацию, а затем распределить выделенные понятия по заранее подготовленной онтологии, т.е. «населить» онтологию [Witte и др., 2010]. Таким образом, OwlExporter не создает онтологию на основе текста, а только расширяет, дополняет уже существующую.

OntoGrid – инструментальная система для автоматизации построения онтологий предметных областей с использованием Grid-технологий и анализа текстов на естественном языке [Гусев и др., 2005].

Данная система оснащена двуязычным лингвистическим процессором для извлечения знаний из текстов на естественном языке. В качестве базы для морфологического анализа используется электронный словарь Д. Уорта [Worth и др., 1970]. Она содержит 3,2 млн. словоформ. Процесс индексации включает 200 правил. «Ключевая лексика» выявляется при помощи анализа распределения слов в тексте. Разработчиками предложен новый метод выявления сверхфразовых единств, образуемых сгущениями лексических единиц определенного типа. Построение семантических сетей текстовых документов осуществляется следующим образом: производится анализ текста при помощи соответствующего компонента системы анализа текста, в качестве формализма для представления смысла текста в ней используются семантические Q-сети [Загоруйко и др., 2004]. База лингвистических знаний системы анализа текста представляет собой набор элементарных и составных словосочетаний предметной области. Такую базу условно можно разделить на базу реализаций элементарных отношений (БРО) и набор критичных фрагментов (НКФ), по которым определяется, какие элементы онтологии затрагиваются в данном тексте. Далее происходит создание и развитие онтологии в GRID-сети. Для представления структуры онтологии используется известный стандарт OWL.

Кроме того, было найдено три информационные системы, выполняющих функции преобразования моделей [Сухов, 2012].

Язык трансформации ATL (ATLAS Transformation Language) является частью архитектуры управления моделью ATLAS [Bezivin, 2005]. ATL – язык, позволяющий описывать трансформации любой исходной модели в указанную целевую модель.

GReAT (Graph REwriting And Transformation) – язык описания преобразований модели, базирующийся на подходе тройных трансформаций графа [14]. Трансформация, описанная на этом языке, представляет собой набор упорядоченных правил перезаписи графа, которые применяются к входной модели и в результате создают выходную модель.

VIATRA – основанный на правилах и паттернах, язык преобразования для управления графовыми моделями, который комбинирует в единую парадигму спецификации два подхода: математический формализм, основанный на правилах трансформации графа, для описания моделей и абстрактные конечные автоматы, предназначенные для описания потока управления [Karypis и др., 1998].

Описанные выше программные системы либо помогают создавать онтологии, либо преобразуют модели (графы) к заданному виду, т.е. нет системы, которая объединяла бы в себе эти функции. Кроме того, не было найдено информационной системы, которая оценивала бы схожесть преобразованной модели с неким образцом.

2. Описание предлагаемого подхода

Процесс выбора подходящего для решения поставленной задачи DSL состоит из шести последовательных этапов, которые должны быть обязательно выполнены (рисунок 1).

Сначала обрабатывается корпус документов, относящихся к предметной области, с целью выделения ключевых понятий. На втором этапе проводится повторная обработка корпуса документов и задаются связи между концептами. Концепты и связи между ними формируют семантическую сеть. Третьим этапом является устранение синонимии среди концептов. На четвертом шаге сжатая семантическая сеть трансформируется в базовую онтологию при помощи алгоритма огрубления графов. После этого специалист уточняет онтологию (если в этом есть необходимость). И, наконец, из хранилища выбираются DSL, наиболее точно описывающие рассматриваемую предметную область.

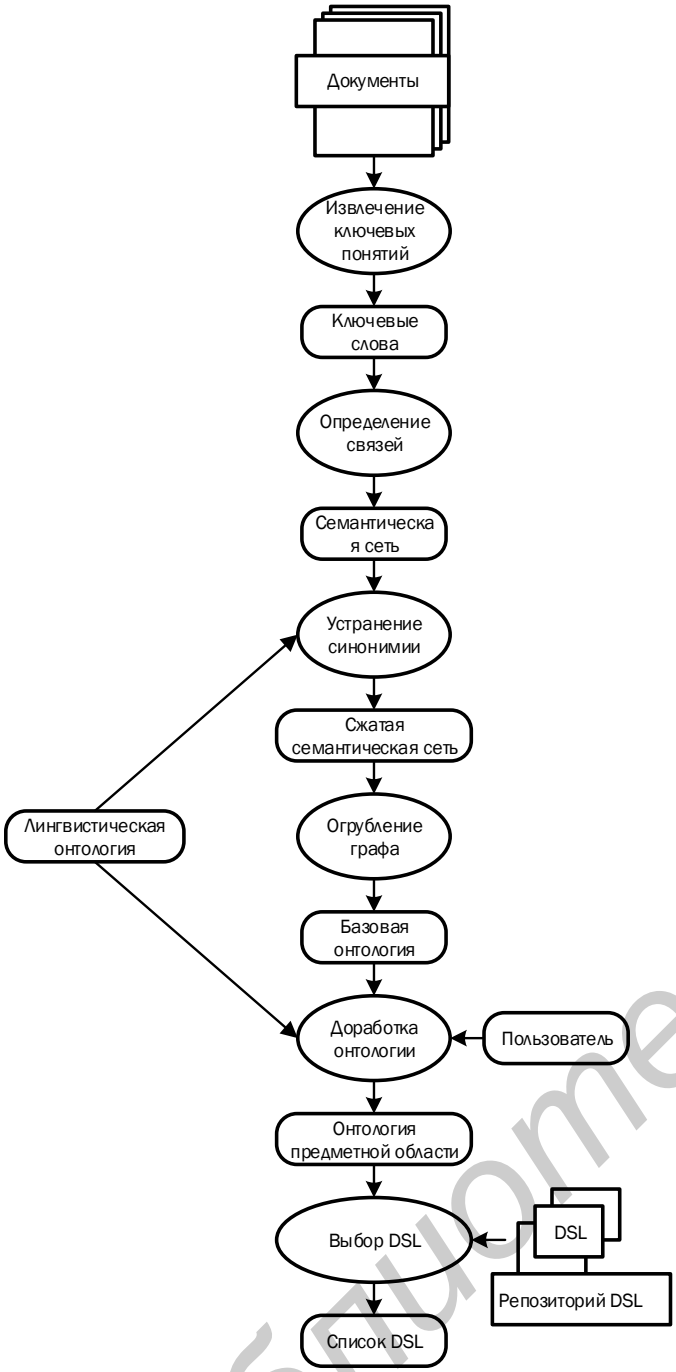

Рисунок 1 – Процесс подбора DSL

2.1. Извлечение ключевых понятий

Одним из наиболее распространенных способов структурирования информации являются онтологии.

Формально онтология определяется как

$$O = <X, R, F>,$$

где X — конечное множество понятий предметной области, R — конечное множество отношений между понятиями, F — конечное множество функций интерпретации.

В рамках данной статьи рассмотрим задание множества понятий и множества отношений.

Будем считать, что основными терминами документа являются его ключевые слова, а именно существительные. Исследования, связанные с алгоритмами выделения ключевых слов документа, активно проводятся в рамках решения задач поиска документов. Все они базируются на частотных законах, открытых лингвистом и филологом Джорджем Ципфом (George Kingsley Zipf). Первый из законов гласит, что произведение вероятности обнаружения слова в тексте на ранг частоты, есть число постоянное. Второй говорит, что частота и количество слов, входящих в текст с этой частотой, также имеют зависимость.

На данный момент для поиска ключевых слов используются закономерности, открытые Ципфом в чистом виде (TF-IDF), либо алгоритмы LSI (latent semantic indexing). В рамках текущего исследования остановимся на первом варианте, как на наиболее простом в реализации. Лингвистическая обработка будет проводиться программными средствами Aot.ru.

Рассмотрим построение онтологии на примере процесса сдачи экзамена. Предположим, с помощью частотного анализа были выделены следующие понятия: программирование, студент, преподаватель, учитель, дискретная математика (рисунок. 2).

Рисунок 2 – Ключевые понятия в процессе сдачи экзаменов

2.2. Определение связей

В результате проведения частотного анализа получим множество несвязанных между собой вершин. Теперь необходимо задать множество связей, т.е. достроить имеющийся несвязный граф до семантической сети.

Семантический граф представляет собой взвешенный граф, вершинами которого являются термины проанализированных документов. Наличие ребра между двумя вершинами означает тот факт, что термины семантически связаны между собой, вес ребра является численным значением семантической близости двух терминов, которые соединяет данное ребро [Гринева и др., 2009].

Величина близости онтологических концептов оценивается по ряду мер:

1. Сходство по Жаккару [Real и др., 1996]:

$$K_J = \frac{c}{a+b-c} \qquad (1)$$

Коэффициент Жаккара – бинарная мера сходства, предложенная Полем Жаккаром в 1901 г., где a – количество упоминаний первого понятия, b – количество упоминаний второго понятия, c –

количество совместного упоминания понятий (понятия встречаются в одном контексте).

2. Взаимная информация [Mutual Information]:

$$MI = \sum_{u=\{0,1\}} \sum_{v=\{0,1\}} P(u,v) log_2 \frac{P(u,v)}{P(u)P(v)} \approx \approx \sum_{u=\{0,1\}} \sum_{v=\{0,1\}} \frac{(u,v)}{N} log_2 \frac{(u,v)}{(u)(v)} N \quad (2)$$

где *u, v* – понятия, встречающиеся в документе; (u) – количество употребления понятия *u*, (*v*) – количество употреблений понятия (*v*),– количество совместного употребления понятий (*u, v*).

При вычислении сходства по частотам совместной встречаемости слов u и v в некотором контексте также использовалась как MI(*u*, *v*), так и точечная взаимная информация PMI [9]:

$$PMI(u,v) = p\left(\frac{(u,v)}{p(u)p(v)}\right) \quad (3)$$

После подсчета оценок близости концептов различными способами производится усреднение этих оценок [Мисуно и др., 2005]. На основе средних оценок в к вершинам добавляются связи. В итоге получаем семантическую сеть (рисунок 3).

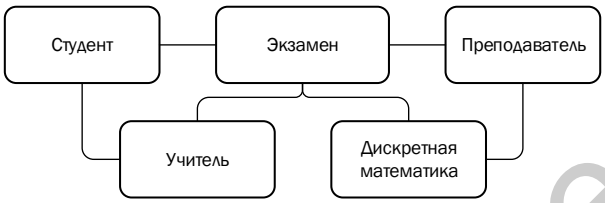

Рисунок 3 – Семантическая сеть понятий процесса сдачи экзамена

2.3. Устранение синонимии

Понятия в тексте могут обозначаться по-разному. В ходе частотного анализа названные иным образом понятия были выделены как отдельные концепты. Таким образом, объединить в одну вершину понятия, являющиеся синонимами.

Синонимия понятий определяется на основе лингвистической онтологии. Мы будем использовать WordNet, семантическую сеть, разработанную в Принстонском университете. Её словарь состоит из 4 сетей для основных частей речи – существительных, глаголов, прилагательных и наречий. Базовая словарная единица – синонимический ряд (синсет), объединяющий слова со схожим значением и, по сути, являющийся узлом сети. Синсеты связываются различными семантическими отношениями, такими, как: гипероним (завтрак → приём пищи), гипоним (приём пищи → обед), has-member (факультет → профессор), member-of (пилот → экипаж), мероним (стол → ножка), антоним (лидер → последователь). Используются различные алгоритмы, например, алгоритмы, учитывающие расстояние между концептуальными категориями слов, учитывающие иерархическую структуру онтологии WordNet.

В рассматриваемом примере понятия «преподаватель» и «учитель» являются синонимами (согласно лингвистической онтологии), поэтому заменяются одной вершиной «преподаватель» (рисунок 4). Вновь созданная вершина содержит связи обеих синонимичных вершин.

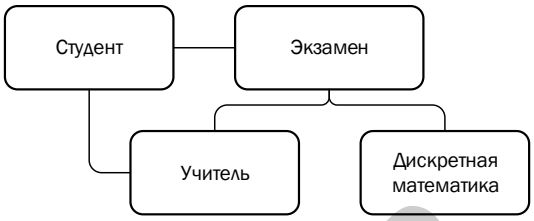

Рисунок 4 – Сжатая семантическая сеть понятий процесса сдачи экзамена

2.4. Огрубление графа

Далее необходимо привести построенную семантическую сеть к онтологической модели. В общей постанове эту задачу следует отнести к задаче огрубления графа [Karypis и др., 1998].

Классические методы решения задачи огрубления графа основаны на итерационном стягивании смежных узлов графа *G*α в узлы графа *G*α+1, где α = 0,1,2,.. – номер итерации, *G*(0) = *G*(O). В результате этого процесса ребро между двумя вершинами графа *G*α удаляется и создается мультиузел графа *G*α+1, объединяющий оба стягиваемых узла [Карпенко, 2010].

Стягиваемые вершины должны иметь общего «родителя». Когда вершины заменяются на одну, то значения, которые имели эти вершины, заменяются значением вершины-родителя из лингвистической онтологии.

В нашем примере «программирование» и «дискретная математика» были объединены в мультиузел «дисциплина» (рисунок 5).

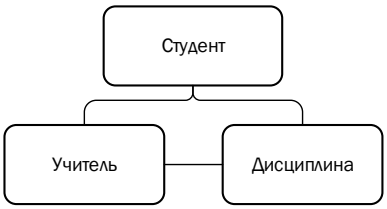

Рисунок 5 – Базовая онтология понятий процесса сдачи экзамена

2.5. Доработка онтологии

В итоге получаем базовую онтологию, представляющую критерии, по которым будет подбираться искомый DSL. Но у построенной модели имеется ряд недостатков:

1. Не отражена семантика связей.
2. Могут присутствовать концепты, которые не важны с точки зрения рассматриваемой задачи (но были выделены в ходе анализа).
3. Могут отсутствовать важные для решения задачи концепты.

Поэтому, пользователю предлагается отредактировать базовую онтологию. Он может задать семантику связей, добавить или удалить концепты по своему усмотрению. Очевидно, чем полнее онтологическая модель, тем точнее будет подобран DSL.

Например, специалист заменил концепт «дисциплина» на «экзамен», удалил связь между студентом и преподавателем, и добавил семантическую значимость связям (студент сдает экзамен, а преподаватель принимает экзамен). Результат действий показан на рисунке 6.

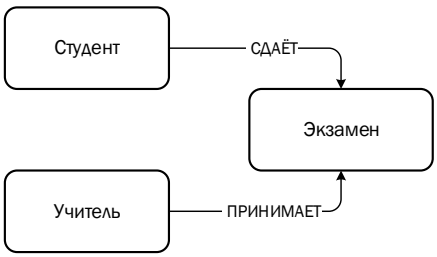

Рисунок 6 – Онтология предметной области «процесс сдачи экзамена»

2.6. Оценка соответствия DSL и созданной онтологии

Сопоставление онтологий сводится к вычислению или выявлению связей или соответствий между понятиями разных онтологий или модулей онтологий, используя различные методы (лексические, структурные и т.д.). Результатом сопоставления является множество соответствий между семантически связанными сущностями.

Мерой соответствия DSL составленной онтологии является показатель изоморфизма графов (насколько графы изоморфны по отношению друг к другу), но с учетом того, что важно рассматривать соответствие не только вершин графов и связей между ними, но и их семантики.

Два графа ($V1$; $E1$; $g1$) и ($V2$; $E2$; $g2$) изоморфны, если существует взаимно однозначное соответствие:

$$f1 : V1 \to V2 \text{ и } f2 : E1 \to E2 \qquad (4)$$

таким образом, для каждой вершины

$$\begin{cases} a \in E1 \\ g1(a) = x - y \end{cases} \qquad (5)$$

тогда и только тогда

$$g2[f2(a)] = f1(x) - f1(y). \qquad (6)$$

Далеко не всегда возможно встретить полную изоморфность двух графов. В том случае, если графы не изоморфны, а только похожи, необходимо проверить существует ли соответствие:

$$f: V1 \to V2, \qquad (7)$$

которое представляет соответствие вершин.

Следует отметить, что в нашем случае графы, вероятнее всего, будут неизоморфны. Очевидно, чем больше «показатель изоморфности» (например, таким показателем может стать число вершин, для которых не удалось установить соответствие, отсутствие или присутствие «лишних» связей и т.п.), тем точнее конкретный DSL описывает предметную область.

Заключение

В рамках данной статьи рассмотрена проблема подбора DSL для решения определенной задачи.

В дальнейшем планируется увеличить количество методов, при использовании которых устанавливаются связи в онтологической модели, для увеличения точности средневзвешенной оценки близости понятий. Кроме того, планируется рассмотреть возможность сравнения DSL не только на одном уровне, но и на разных (добавить возможность сравнивать иерархические структуры).

Работа выполнена при поддержке Научного фонда НИУ ВШЭ по программе софинансирования грантов РФФИ (проект № 13-09-0143).

Библиографический список

[Гринева и др., 2009] Гринева М., Гринев М., Лизоркин Д. Анализ текстовых документов для извлечения тематически сгруппированных ключевых терминов. [Электронный ресурс]. URL: http://citforum.ru/database/articles/kw_extraction/2.shtml#3.3.

[Гусев и др., 2005] Гусев В., Завертайлов А., Загоруйко Н. и др. Система “Онтогрид” для построения онтологий. [Электронный ресурс]. URL: http://www.dialog-21.ru/Archive/2005/Zagoruiko%20Gusev%20Zavertailov/ZagoruykoNG.htm.

[Загоруйко и др., 2004] Загоруйко Н., Налётов А., Соколова А. и др. Формирование базы лексических функций и других отношений для онтологии предметной области. [Электронный ресурс]. URL: http://www.dialog-21.ru/Archive/2004/Zagorujko.htmM. Young, The Technical Writer’s Handbook. Mill Valley, CA: University Science, 1989.

[Карпенко, 2010] Карпенко А. Оценка релевантности документов онтологической базы знаний. [Электронный ресурс]. Режим доступа: http://technomag.edu.ru/doc/157379.html.

[Лядова, 2010] Лядова Л.Н. Многоуровневые модели и языки DSL как основа создания интеллектуальных CASE-систем. [Электронный ресурс]. URL: http://www.hse.ru/data/2010/03/30/1217475675/Lyadova_LN_2.pdf.

[Мисуно и др., 2005] Мисуно И., Рачковский Д., Слипченко С. Векторные и распределенные представления, отражающие меру семантической связи слов. [Электронный ресурс]. URL: http://www.immsp.kiev.ua/publications/articles/2005/2005_3/Misuno_03_2005.pdf.

[Сухов, 2012] Сухов А. О. Методы трансформации визуальных моделей. [Электронный ресурс]. Режим доступа: http://www.hse.ru/pubs/share/direct/document/68390345.

[Bezivin, 2005] Bezivin J. An Introduction to the ATLAS Model Management Architecture. [Online]. Available: http://www.ie.inf.uc3m.es/grupo/docencia/reglada/ASDM/Bezivin05b.pdf.

[Mutual Information] Mutual Information [Online]. Available: http://cats.lse.ac.uk/homepages/liam/st418/mutual-information.pdf.

[Karypis и др., 1998] Karypis G., Kumar V. Multilevel k-way Partitioning Scheme for Irregular Graphs // Journal of Parallel and Distributed Computing,. 1998. Pp. 96-129.

[Real и др., 1996] Real R., Vargas J., The Probabilistic Basis of Jaccard's Index of Similarity [Online]. Available: http://sysbio.oxfordjournals.org/content/45/3/380.full.pdf.

[Sukhov and Lyadova, 2012] Sukhov A.O., Lyadova L.N. MetaLanguage: a Tool for Creating Visual Domain-Specific Modeling Languages // Proceedings of the 6th Spring/Summer Young

Researchers' Colloquium on Software Engineering, SYRCoSE 2012, Пермь: Институт системного программирования Российской академии наук, 2012. Pp. 42-53.

[Witte и др., 2010] Witte R., Khamis N., and Rilling J. Flexible Ontology Population from Text: The OwlExporter // Dept. of Comp. Science and Software Eng. Concordia University, Montreal, Canada. [Online]. Available: http://www.lrec-conf.org/proceedings/lrec2010/pdf/932_Paper.pdf.

[Worth и др., 1970] Worth D., Kozak A., Johnson D. Russian Derivational Dictionary / New York, NY: American Elsevier Publishing Company Inc, 1970.

AN APPROACH TO THE SELECTION OF DSL BASED ON CORPUS OF DOMAIN-SPECIFIC DOCUMENTS

Valeev M.T., Elokhov E.S., Uzunova E.N., Yugov A.S.

National Research University Higher School of Economics, Perm, Russia

mt.vallev.1992@gmail.com

eugene.yelokhov@gmail.com

palgonuri@gmail.com

yugovas@live.ru

Today many problems that are dedicated to a particular problem domain can be solved using DSL. Thus to use DSL it must be created or it can be selected from existing ones. Creating a completely new DSL in most cases requires high financial and time costs. Selecting an appropriate existing DSL is an intensive task because such actions like walking through every DSL and deciding if current DSL can handle the problem are done manually. This problem appears because there are no DSL repository and no tools for matching suitable DSL with specific task. This paper observes an approach for implementing an automated detection of requirements for DSL (ontology-based structure) and automated DSL matching for specific task.

Introduction

Nowadays metamodeling and DSL-based technologies (DSL – Domain Specific Language) are widely used in information system developing. DSL is created for solving some specific problem. Almost every arising problem is similar to the one that was solved before. In this case it means that a suitable DSL was already implemented or an implemented DSL does not fully meet the requirements. Therefore, you can either find a ready-to-use DSL or complete and configure a DSL implemented earlier. This requires less costs rather than developing a completely new DSL.

This paper shows generating process of requirements ontology based on domain-specific documents and how a particular DSL meets given requirements.

Main Part

The suggested approach of the DSL selection process consists of six stages that can be described as a series of sequential operations which should be implemented.

Firstly, a corpus of documents is processed. As a result, the key words (concepts related to specific domain) are retrieved. Secondly, when re-viewing the document, the relations between concepts are built. These concepts and relations form a semantic network. The next step is to eliminate synonymy (to merge nodes containing synonymic concepts). In order to achieve this, a linguistic ontology is used. After that, it is necessary to transform ~~–c~~ontracted" semantic network into ontology model, using the graph coarsening algorithm with implementing linguistic ontologies. The next step is to qualify the ontology model by a specialist. This step includes concepts editing and relations marking semantically.

When the ontology is complete, i.e. it meets user requirements, DSLs are taken from the repository, and the measures of DSLs correspondence to ontology requirements are calculated.

Conclusion

In this paper a problem of matching a suitable DSL for specific task was observed.

The requirements for DSL are based on domain documents analysis. Requirements are formed as ontological model which is generated in two steps: defining concepts using frequency analysis of terms found and defining relations based on average weighted score obtained using Jaccard index and mutual information index.

The second step of DSL matching is comparison of DSL's that was implemented earlier with ontology based on domain documents analysis. The core of this comparison is the method of determining graphs' isomorphism and semantic match is controlled by linguistic ontology.