АЛГОРИТМИЧЕСКОЕ ОБЕСПЕЧЕНИЕ СЕМАНТИЧЕСКОГО АНАЛИЗА ТЕКСТА ТЕХНИЧЕСКОГО ЗАДАНИЯ

Ю.А. Орлова

Волгоградский государственный технический университет, кафедра "САПР и ПК"
Пр. Ленина 28, г. Волгоград, Россия
телефон: + (8442) 248108; факс: + (8442) 248100; e-mail: Yulia.Orlova@gmail.com

В работе рассматривается алгоритмическое обеспечение анализа текста технического задания и автоматического построения модели программного обеспечения в виде диаграмм потоков данных.

Ключевые слова – алгоритмические обеспечение, семантический анализ текста, модель программного обеспечения, диаграммы потоков данных.

Разработка и анализ технической документации требует от лиц, занимающихся проектированием программного обеспечения семантической обработки большого объема технического текста, глубокого знания предметной области и навыков в проектировании. Трудоемкость процесса анализа текста приводит к необходимости его автоматизирования. Однако необычайная сложность проблемы синтеза и анализа семантики технического текста, для решения которой необходимо использовать сембиоз методов искусственного интеллекта, прикладной лингвистики, психологии и т.п., приводит к тому, что она до сих пор не решена.

В данной работе мы пытаемся автоматизировать начальный этап проектирования программного обеспечения - семантический анализ текста технического задания.

Общий алгоритм семантического анализа текста технического задания состоит из следующих блоков: предварительная обработки текста, синтаксический и семантический анализ и построение модели программного обеспечения.

Предварительная обработка текста осуществляется с использованием аппарата конечных автоматов

Входные символы конечного автомата: c1 - пустое пространство, c2 - пробел, c3 - новая строка, c4 - конец текста, c5 - '1'..'9', c6 -'П', c0 - любой другой символ.

Промежуточные состояния автомата: a1 - начало разбора номера раздела, a2 - последовательность символов – текст, a3 - последовательность символов – нумерация, a4 - начало разбора названия раздела, a5 - последовательность символов – название раздела, a6 - начало разбора текста раздела или приложения, a7 - последовательность символов – продолжение текста раздела или приложения, a8 - начало разбора названия приложения, a9 - последовательность символов – название приложения, a0 - конец ТЗ.

В ходе работы конечного автомата символы, поступающие на его вход, накапливаются в буфере. В определенных состояниях конечного автомата осуществляется запись текущего содержимого буфера в одну из таблиц, после чего буфер опустошается. Работа автомата продолжается до достижения конечного состояния.

На выходе алгоритма предварительной обработки текста формируется набор таблиц: разделов, предложений и лексем. После этого полученные таблицы поступают на вход алгоритма семантического анализа (рис. 1).

Рис.1. Общий алгоритм семантического анализа текста.

Семантический анализ текста производится на основе разработанной нечеткой атрибутной грамматики над фреймовой структурой текста ТЗ:

1. Каждая лингвистическая переменная технического задания подвергается разбору, в результате чего получается лингвистическое дерево, конечными вершинами которого являются нечеткие переменные.

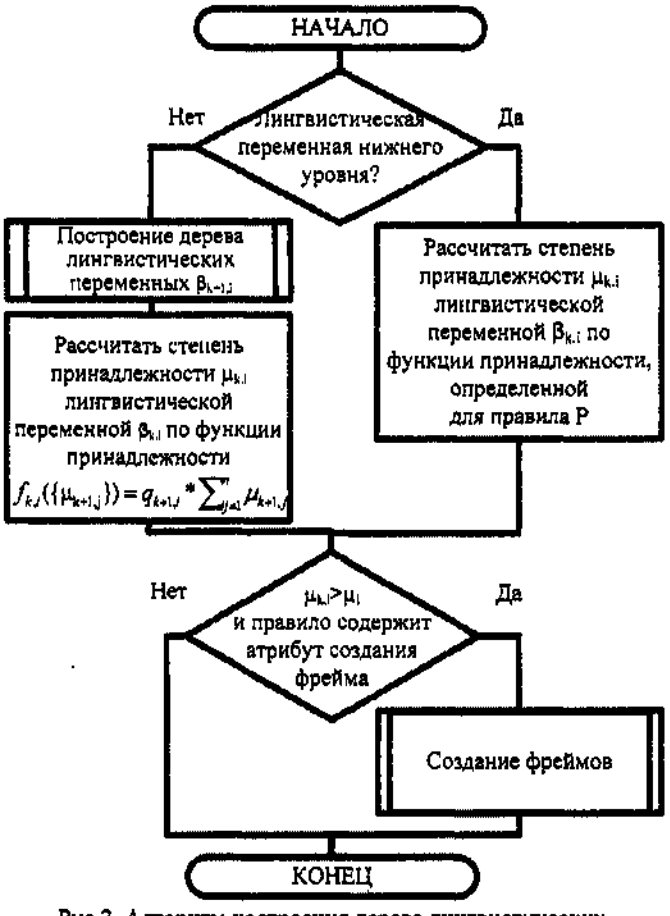


Рис.2. Алгоритм построения дерева лингвистических переменных βk,i.

2. Нечетким переменным на конечных вершинах дерева назначается их смысл и затем с помощью системы правил P и соответствующих функций принадлежности fk,i определяется смысл лингвистической переменной, соответствующей левой части правила (рис. 2).

Правила верхнего уровня служат для разбора разделов верхнего уровня. Правила для разбора разделов состоят из двух частей: первая часть служит для разбора названия раздела; вторая часть служит для разбора текстового содержимого раздела.

Для некоторой лингвистической переменной βk,i значение функции принадлежности:

$$\mu k,i=fk,i(\mu k+1,1,\mu k+1,2,\ldots,\mu k+1,n) \qquad (1)$$

где конкретное значение μk,i – степень принадлежности лингвистической переменной βk,i. Первоначально будем говорить, что все лингвистические переменные нижнего уровня вносят одинаковый вклад в значение функции принадлежности, поэтому можно говорить, что функция принадлежности лингвистической переменной βk,i:

$$f_{k,i}(\{\mu_{k+1,j}\}) = q_{k+1,j} * \sum_{j=1}^{n} \mu_{k+1,j} \qquad (2)$$

где μk+1,j - степень принадлежности лингвистической переменной βk+1,j; qk+1,j = 1/n – вклад степеней принадлежности в значение функции. На нижнем уровне функции принадлежности определены.

Вычисленная μk,i сравнивается с μl, являющейся предельным значением степени принадлежности. Если μk,i>μl, и в правилах указаны синтаксические или семантические атрибуты, то создаются фреймы и слоты, в которые помещается текст из соответствующей лингвистической переменной.

3. После этого дерево урезают так, чтобы вычисленные лингвистические переменные оказались конечными вершинами оставшегося поддерева.

Этот процесс повторяется до тех пор, пока не будет вычислен смысл лингвистической переменной, соответствующей корню исходного дерева. Основное назначение описанной процедуры состоит в том, чтобы связать смысл лингвистической переменной со смыслом составляющих ее нечетких переменных посредством грамматики.

В ходе разбора синтаксический и морфологический анализ производятся только в том случае, если имеется необходимость, что значительно сокращает время выполнения семантического анализа. Если в правиле грамматики встретился терминал, имеющий синтаксический атрибут, то запускается механизм синтаксического анализа для текущего разбираемого предложения.

После создания дерева лингвистических переменных начинается построение фреймового описания технического задания. Для этого используется информация о фреймах и названиях слотов, которая содержится в атрибутах символов грамматики.

Полученная фреймовая структура содержит значимую информацию о системе: сведения о входах и выходах, функциях и ограничениях. Для каждой функции также выделяются входы и выходы. Это позволяет на основе фреймовой структуры получить диаграммы потоков данных системы, которая описана в техническом задании. Алгоритмы создания фреймов и построения диаграмм потоков данных представлены на рисунках 3 и 4.

На основе представленных выше алгоритмов разработана автоматизированная система семантического анализа текста технического задания "Семантика ТЗ", которая состоит из следующих подсистем: "Хранение документов", "Интерфейс", "Предварительная обработка текста", "Синтаксический анализ", "Семантический анализ", "Построение диаграмм потоков данных".

Проект разработан на платформе Microsoft.NET Framework (язык разработки C#). Таблицы разделов хранятся в формате XML, а их визуальное представление возможно с использованием XSL-преобразования. Полученное при семантическом анализе фреймовое описание также сохраняется в формате XML. Построение диаграмм потоков данных осуществляется с помощью взаимодействия системы с программой MS Visio.

Была проведена комплексная проверка предложенных алгоритмов помощью АС "СемантикаТЗ".

Рассмотрим пример работы системы. Фрагмент фреймовой структуры для автоматизированной системы расчета локальной сети представлен на рисунке 5, результат

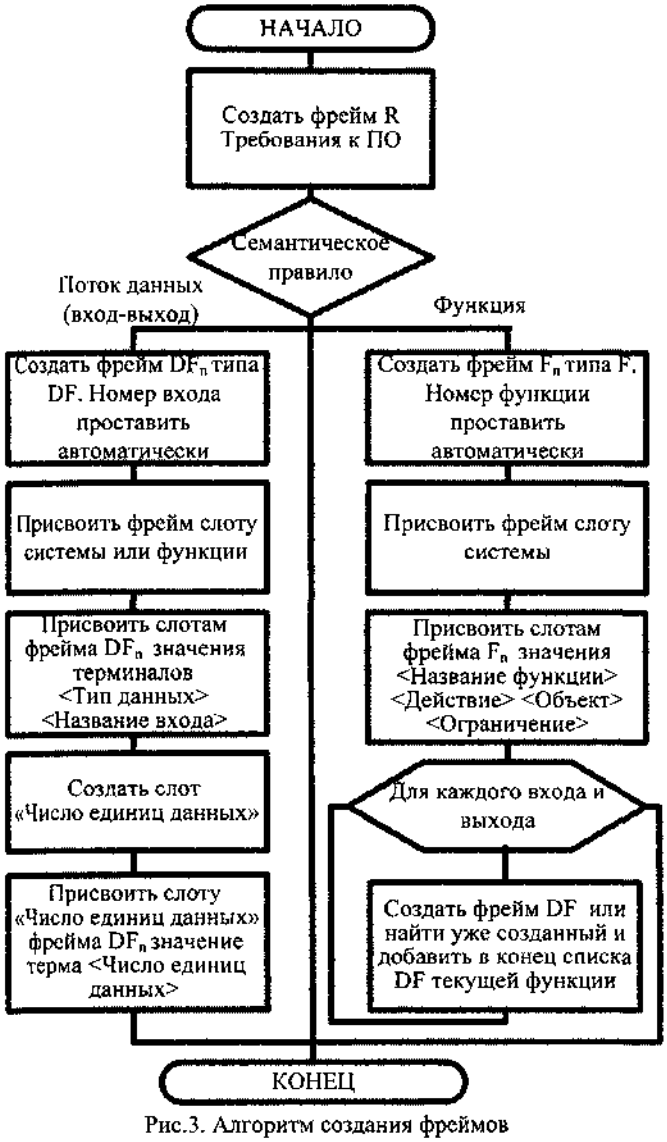


Рис.3. Алгоритм создания фреймов

работы системы в виде фрагмента диаграмм потоков данных, синтезированной на основе фреймовой структуры, представлен на рисунке 6.

Эффективность применения алгиритмов тем выше, чем больше число функциональных единиц, представленных в техническом задании. Время анализа ТЗ с использованием системы сокращается на 40-70%.

Таким образом, разработанная автоматизированная система позволяет повысить эффективность проектирования программного обеспечения на ранних этапах за счет сокращения времени работы над техническим заданием и увеличения качества получаемого результата.

Основные результаты работы:

1. Разработано алгоритмическое обеспечение семантического анализа текста технического задания: предварительная обработка текста, синтаксический, семантический анализ и построение модели программного обеспечения. Предварительная обработка текста осуществляется с использованием аппарата конечных автоматов. Семантический анализ текста производится на основе разработанной нечеткой атрибутной

Рис.4. Алгоритм построения диаграмм

грамматики. Для вычисления смысла лингвистической переменной и построения нечеткого вывода использовались функции принадлежности. Синтаксический и морфологический анализ производятся только в том случае, если имеется необходимость, что значительно сокращает время выполнения семантического анализа. На основе дерева лингвистических переменных и семантических атрибутов создается фреймовое описание системы и осуществляется построение модели программного обеспечения в виде диаграмм потоков данных.

2. Разработанные формализмы, методика и алгоритмы реализованы в виде системы автоматизации начального этапа проектирования программного обеспечения "СемантикаТЗ" на платформе Microsoft .NET Framework 2.0с использованием визуальной среды программирования Visual Studio 2005. Построение диаграмм потоков данных осуществляется в MS Visio.

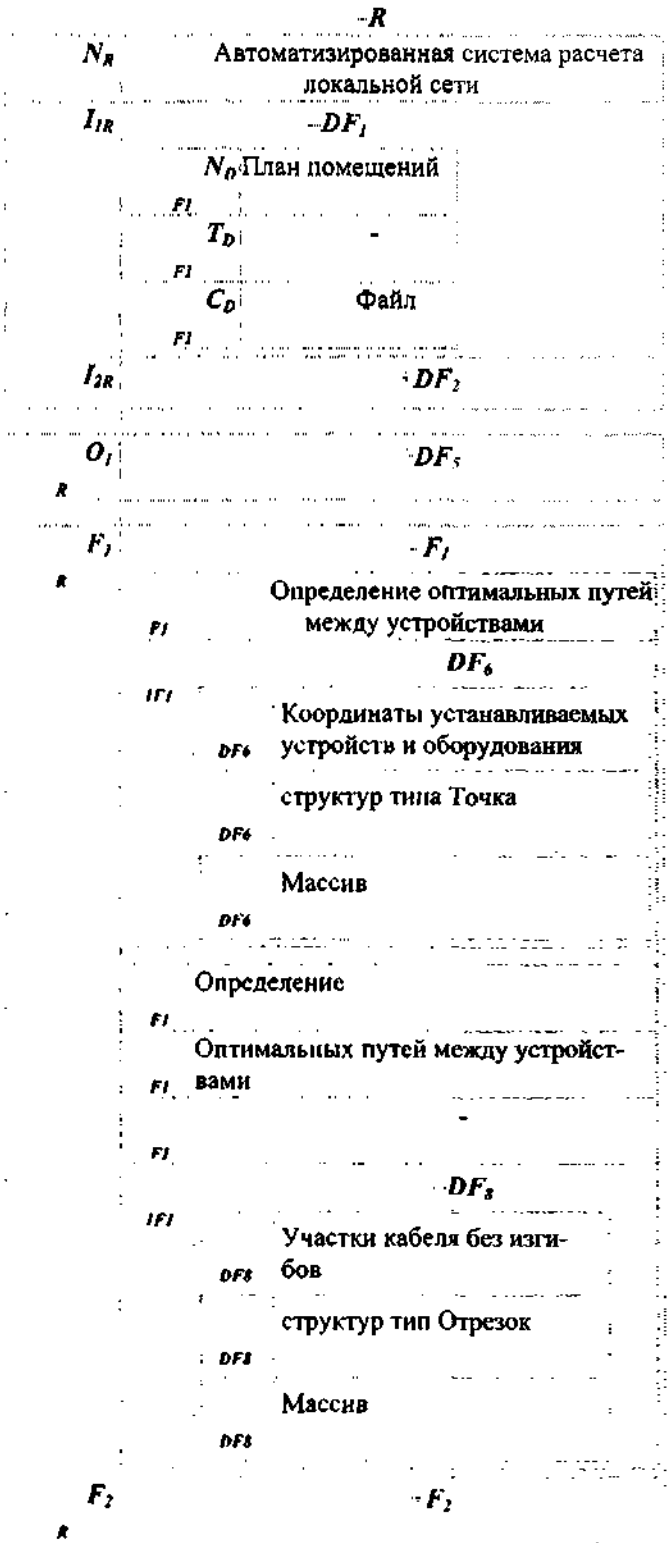
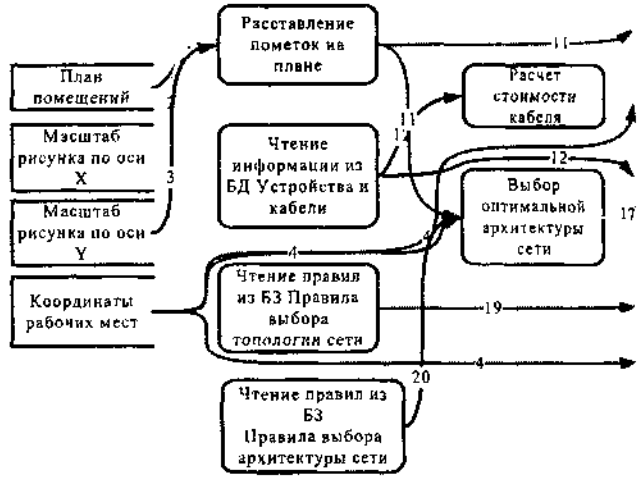


Рисунок 5. Фрагмент заполненной фреймовой структуры

Проведено исследование разработанной программной среды при проектировании программного обеспечения: систем расчета локальной сети, диспетчерского контроля и принятия решений. Результаты исследования позволяют сделать вывод, что разработанные формализмы,



ОБОЗНАЧЕНИЯ ПОТОКОВ ДАННЫХ:

1 - План помещений
3 - Масштаб рисунка по оси Y
4 - Координаты рабочих мест
11 - Разметка пространства карты
12 - Цены устройств и кабелей
19 - Правила выбора топологии сети
20 - Правила выбора архитектуры сети

Рисунок 5. Фрагмент диаграммы потоков данных

методика и алгоритмы соответствуют поставленной цели и задачам.

Повышение эффективности заключается в значительном сокращении времени анализа текста технического задания и построения модели программного обеспечения (от 40% до 70% в зависимости от числа функциональных единиц в ТЗ). Также значительно повысился процент обнаружения и исправления ошибок, сделанных на этапе формализации и анализа ТЗ.

В результате использования разработанной системы повышается качество проектирования программного обеспечения за счет автоматизации рутинного труда человека по извлечению полезной информации из стандартного документа и отображению ее в виде модели программного обеспечения.

Литература

[1] Заболеева-Зотова, А.В. Automation of procedures of the semantic text analysis of a technical specification / А.В. Заболеева-Зотова, Ю.А. Орлова // Интеллектуальные системы (INTELS'2008): тр. 8-го междунар. симпозиума, Нижний Новгород, 30 июня - 4 июля 2008 г.: [к 60-летию каф. "Системы автом. упр." МГТУ им. Н.Э.Баумана] / МГТУ им. Н.Э.Баумана, НГТУ им. Р.Е.Алексеева [и др.]. - М., 2008. - С. 245-248.- Англ.

[2] Заболеева-Зотова, А.В. Computer-aided System of Semantic Text Analysis of a Technical Specification / А.В. Заболеева-Зотова, Ю.А. Орлова // Advanced Research in Artificial Intelligence: suppl. to Int. Journal "Information Technologies and Knowledge". - 2008. - Vol. 2, [Int. Book Series "Inform. Science & Comput."; № 2]. - С. 139-145.- Англ.

[3] Заболеева-Зотова, А.В. Computer-aided system of the semantic text analysis of a technical specification / А.В. Заболеева-Зотова, Ю.А. Орлова // Открытое образование: приложение к журналу [по матер. междунар. конференций, Ялта-Гурзуф, 20-30 мая 2008 г.]. - 2008. - Б/н. - С. 103-104.- Англ.