УДК 004.912

РАЗРАБОТКА ЛИНГВИСТИЧЕСКОГО ПАРСЕРА РУССКОГО ЯЗЫКА

В. В. Гаршина, Ю. А. Богоявленская

Воронежский государственный университет

Поступила в редакцию 21.11.2012 г.

Аннотация. Разработка лингвистических парсеров является актуальной задачей современности. В статье собрана общая информация о парсерах, рассмотрены существующие аналоги и представлено собственное решение для русского языка.
Ключевые слова: компьютерная лингвистика, лингвистические парсеры, морфологический анализ, синтаксический анализ, семантический анализ.
Annotation. Nowadays the development of natural language parsers is an actual challenge. This article contains basic data about the parsers, overview of existing analogues and implemented solution for Russian language.
Keywords: computer linguistics, natural language parsers, morphological analysis, syntax analysis, semantic analysis.

ВВЕДЕНИЕ

В компьютерной лингвистике существует много прикладных задач, которые используют результаты работы специального класса программ – *лингвистических парсеров*. К таким задачам можно отнести: анализ/синтез предложения на естественном языке (ЕЯ) с целью формирования/распознавания запроса на искусственном языке (SQL, SPARQL, поисковые запросы и др.), преобразование предложения с одного ЕЯ на другой ЕЯ (системы машинного перевода), задачи семантического анализа текстов – TextMining, семантическая классификация текстовых ресурсов и др.

Лингвистический парсер – комплекс программных модулей, обеспечивающий разбор линейной последовательности лексем (слов) языка исходного текста во внутреннее представление смысла этого предложения. При этом используется многоуровневый анализ предложения на ЕЯ, реализующий морфологический, синтаксический, семантический (иногда и прагматический) языковой уровень. На каждом языковом уровне используются свои структуры данных, которые обрабатываются и формируются соответствующими компонентами лингвистического парсера.

Морфологический анализатор, используя морфологические словари, строит разбор последовательности входящих в анализируемое предложение слов с указанием части речи и морфологических характеристик.

Синтаксичекий анализатор реализует построение дерева зависимостей, в узлах которого стоят слова данного предложения с указанием части речи и грамматических характеристик, а дуги соответствуют специфичным для данного естественного языка отношениям подчинения.

Семантический анализатор также часто использует дерево зависимостей, но в его узлах стоят либо предметные имена, либо слова универсального семантического языка (например, имена таблиц, в которых сосредоточены сведения о данной предметной области, атрибуты таблиц, операторные символы). Дуги соответствуют универсальным отношениям семантического подчинения: аргументное, атрибутивное, конъюнкция, дизъюнкция, равенство, неравенство, больше, меньше, принадлежит, не принадлежит и т. п. [3]

Для использования в лингвистических проектах возникла необходимость разработки собственного парсера русского языка. В связи с этим, был проведен анализ задач, связанных с каждым из уровней лингвистического анализа, предложены свои решения. Языком программирования выбран C#. Для написания базы правил, контролирующих уровни синтаксического и семантического анализа, используется Prolog в среде SWI-Prolog 5.8.3.

1. АНАЛИЗ СУЩЕСТВУЮЩИХ РЕШЕНИЙ ПАРСЕРОВ РУССКОГО ЯЗЫКА

Задача выбора лингвистических, алгоритмических и программных решений для построения парсера русского языка является сложной, в связи с особенностями структуры языка, трудностями в формализации языковых правил. В частности русский язык является флективным, что подразумевает собой словообразование и словоизменение с использованием некоторых аффиксов, выражающих значения одной или чаще нескольких грамматических категорий. Кроме того, порядок слов в русском языке является свободным, что мешает выработать однозначные правила для синтаксического анализа.

Известным лингвистическим парсером для русского языка можно назвать систему «Диалинг» [5]. Эта система базируется на многоуровневом языковом представлении, заимствованном у системы ФРАП. (системы французско-русского машинного перевода, разработанная коллективом лаборатории машинного перевода Всесоюзного центра переводов совместно с коллективом лаборатории машинного перевода МГПИИЯ им М. Тореза).

В качестве примера можно привести для русского языка Парсер грамматики связей (Russian Link Grammar) [6]. Также интерес представляет лингвистический процессор ЭТАП-3 [7], который осуществляет перевод предложения на русском или английском языке на формальный семантический язык UNL. Пример работы ЭТАП-3 представлен на рисунке 1.

Представляет интерес разработка компании Dictum [8] – синтаксический анализатор для русского языка, решающий задачу омонимии, собирая статистику из Интернета. Он также способен разбирать эллиптические конструкции. На рис.2 представлен пример дерева зависимостей, построенного этой программой для предложения «Ломбард несет ответственность за утрату вещей, если не докажет, что утрата произошла вследствие непреодолимой силы» .

Компания PROMT, известная своими разработками в области машинного перевода, разработала PROMT Syntactic and Semantic Analyzer, выполняющий глубокий морфологи-

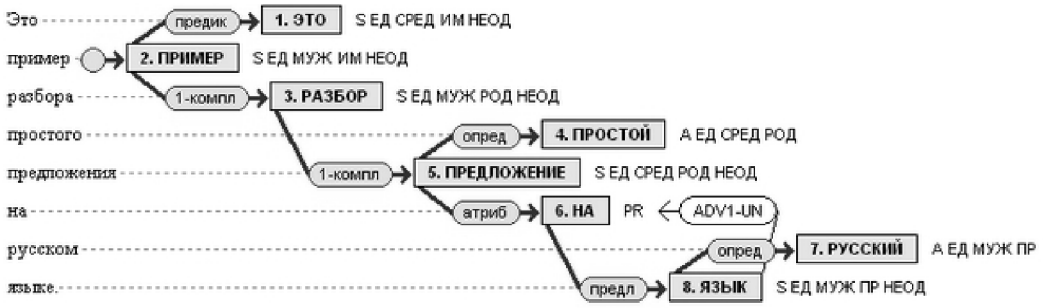

Рис. 1. Дерево зависимостей, полученное в результате синтаксического анализа лингвистическим процессором ЭТАП-3

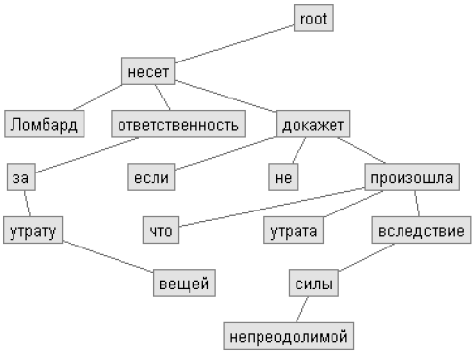

Рис. 2. Дерево зависимостей, построенное синтаксическим анализатором Dictum

ческий, синтаксический и семантический анализ заданного текста на естественном языке. Лингвистическая база данных этого приложения содержит более 30 миллионов словоформ и позволяет вводить новые слова и модели, такие как новые семантические классы или определяемые пользователем синтаксические модели; кроме того, обеспечена поддержка 6 естественных языков. Результат выдается в формате XML, который легко может быть проанализирован. Комплектация продукта также содержит средство визуализации результатов анализа для экспертной оценки [9].

СУЩЕСТВУЮЩИЕ ПОДХОДЫ К РЕАЛИЗАЦИИ МОРФОЛОГИЧЕСКОГО КОМПОНЕНТА

Морфологический компонент лингвистического анализа – это программный модуль, обеспечивающий морфологический анализ лексем исходного языка. Существующие в настоящее время морфологические модели различаются в основном по следующим параметрам.

Во-первых, по результатам работы основанных на них морфологических анализаторов. На вход морфологический анализатор получает словоформу некоторого естественного языка, а на выходе может выдавать все значения грамматических характеристик (род, число, падеж, вид, лицо и т.п.) заданной словоформы, а может просто отвечать на вопрос, принадлежит ли заданная словоформа некоторому естественному языку или нет (в этом случае морфологические анализаторы называют акцепторами).

Во-вторых, морфологические модели могут ориентироваться на полное покрытие лексики (т.е. все лексемы, которые могут обрабатывать программы морфологического уровня, находятся в базе данных) или частичное покрытие лексики (морфологическая модель учитывает возможность появления лексемы, не занесенной в базу данных).

В-третьих, морфологические модели различаются по способу представления и членения словоформ. Существует два основных способа представления лексем.

1) В базе данных хранятся все словоформы всех лексем (возможно, с набором их грамматических характеристик), и каким-то образом определяются словоформы, принадлежащие одной лексеме. Такой способ представления лексем удобен и эффективен для малофлективных языков, в которых различные грамматические категории реализуются, в основном, не с помощью вариации флексий, а некоторым грамматическим способом, например, с помощью предлогов. К малофлективным языкам относится, например, английский язык.

2) В базе данных хранятся основы лексем и списки флексий (возможно, с приписанными им значениями грамматических характеристик), которые присоединяются к основе для получения какой-либо словоформы. Такой способ представления лексем эффективен для флективных языков, в которых различные грамматические категории реализуются путем вариации флексий. Флективным является, например, русский язык. Модели, в которых принят данный способ представления лексем подразделяются еще на две группы: в одной учитываются чисто орфографические основы и флексии, в другой – так называемые псевдоосновы (неизменяемая начальная часть слова) и псевдофлексии (варьируемая при словоизменении конечная часть слова). Выбор того или иного варианта определения основы связан, в основном, с эффективностью реализации и назначением морфологического компонента в целом [3].

АНАЛИЗ ПОДХОДОВ К РЕАЛИЗАЦИИ СИНТАКСИЧЕСКОГО КОМПОНЕНТА

Синтаксический компонент производит синтаксический анализ предложения (parsing). На вход он получает данные морфологического анализа, а на выходе должен построить *дерево зависимостей*, отражающее структуру разобранного предложения. При этом предложение представляется как линейно упорядоченное множество элементов (словоформ), на котором можно задать ориентированное дерево (узлы – элементы множества). Каждая дуга, связывающая пару узлов, интерпретируется как *подчинительная* связь между двумя элементами, направление которой соответствует направлению данной дуги.

Среди способов построения дерева зависимостей выделяют:

• *Модель разбора на составляющие (phrase structure-based parsing*).

• *Модель отношений между словами (dependency parsing)*

Суть dependency parsing состоит в том, что соединение зависимых слов происходит без создания дополнительных узлов. Центром практически любой фразы является глагол (явный

или подразумеваемый). Далее от глагола (действия) можно задавать вопросы: кто делает, что делает, где делает и так далее. Для присоединённых сущностей тоже можно задать вопросы (в первую очередь, вопрос «какой»). Этот метод подходит для языков со свободным порядком слов (например, русского языка), в отличии от *Модели разбора на составляющие,* которая реализует разбор на основе грамматики Ноама Хомского и используется для языков с регулярной структурой.

РЕШЕНИЯ РЕАЛИЗАЦИИ СЕМАНТИЧЕСКОГО КОМПОНЕНТА

Семантический компонент может быть реализован на основе онтологий. В этом случае выделяется три компонента знаний: онтология, включающая в себя *понятия и отношения ПО*; предметный словарь (тезаурус), содержащий термины, с помощью которых в тексте могут представляться понятия и отношения онтологии; информационное наполнение системы или база данных.

В рамках реализации такого подхода необходимо:

– описать понятия (классы), которым соответствуют текстовые ресурсы;

– определить формальную структуру содержания для каждого класса текстовых ресурсов;

– задать схемы фактов, задающие правила извлечения содержательных объектов из текста [11].

Другим подходом к решению задачи семантического анализа является использования формальных семантик. В этом случае значение предложения представляется с помощью формулы лямбда-исчисления.

Для лингвистических целей определяются l-оператор и две операции (a-конверсия и b-конверсия). Синтаксически l-оператор работает так же, как работают кванторы всеобщности и существования: l ставится перед переменной, после чего эта переменная считается связанной во всем подкванторном выражении. Каждое вхождение переменной, которая связана оператором l, является дыркой, ждущей заполнения формулой. Таким образом, переменная, связанная оператором l, эксплицитно помечает места недостающей информации, которые нужно заполнить. Операция, называемая b-конверсией, осуществляет необходимые подстановки.

Общий механизм работы системы на лямбда-исчислении для одного предложения выглядит так:

1. Всем словам приписывается по формуле, составленной по законам лямбда-исчисления, которые в результате составляют одну большую формулу;

2. Последовательно применяя b-конверсию и другие законы логики первого порядка, упрощаем формулу;

3. В результате должна получиться формула без l-операторов и операторов конкатенации.

Формальные семантики позволяют быстро перейти от языкового выражения к его логической структуре и референтам высказывания, но на очень ограниченном "фрагменте" языка [5].

Существует так же подход, называемый латентно-семантическим анализом (LSA, Latent Symantic Analysis). В этом подходе значение слов оценивается путем статистической обработки больших корпусов текстов, включающих в себя набор взаимных ограничений, благодаря которому можно определить семантическую идентичность двух и более слов. Для определения этих ограничений применяются методы линейной алгебры, в частности, сингулярное разложение [12].

2. РЕАЛИЗАЦИЯ УРОВНЕЙ ЛИНВИСТИЧЕСКОЙ ОБРАБОТКИ ДЛЯ ПАРСЕРА РУССКОГО ЯЗЫКА

МОРФОЛОГИЧЕСКИЙ АНАЛИЗАТОР

В данной работе используется морфологический анализатор, который строится на основе морфологических моделей с частичным покрытием лексики. Поэтому выбран для него морфологический словарь, хранящий только основы лексем и списки флексий – электронный морфологический словарь системы Диалинг [10] для русского языка. Этот словарь базируется на грамматическом словаре А. А. Зализняка [4] и включает на данный момент 161 тыс. лемм. – базовых словоформ русского языка с их полным морфологическим описанием.

В используемой морфологической модели учитываются значения грамматических характеристик лексем, с каждой лексемой связаны: синтаксический класс (часть речи), словоизменительный (парадигматический) класс и значения грамматических категорий, или грамматических переменных (ГП), соответствующих синтаксическому классу. Различаются свобод-

ные и связанные ГП. Связанные ГП – ГП, присущие лексеме в целом (всем ее словоформам), например, одушевленность и род для существительных. Свободные ГП – совокупность ГП, по которым лексема изменяется, например, число и падеж для существительных.

В один синтаксический класс объединяются лексемы, имеющие

- общий набор ГП;
- общий набор свободных ГП;
- общее множество значений ГП;
- общие синтаксические функции;

В грамматике (русского языка) выделяются следующие синтаксические классы, с которыми связаны следующие ГП (для классов неизменяемых лексем ГП не указаны).

Существительные. ГП – одушевлённость, род, число, падеж. Свободные ГП – число, падеж.

Прилагательные. ГП – одушевлённость, род, число, падеж, степень. Свободные ГП для полных форм – одушевленность, род, число, падеж. Свободные ГП для кратких форм – род, число. Свободные ГП для сравнительной степени – степень.

Глаголы. ГП личных форм глагола – возвратность, вид, наклонение-время, лицо, род, число; кроме того, переходные глаголы имеют формы страдательного залога. Свободные ГП личных форм глагола – наклонение-время, лицо, род, число, залог. Причастия и деепричастия являются глагольными формами и входят в парадигму глагола. ГП причастий – возвратность, вид, время, залог, одушевленность, род, число, падеж. Парадигма причастий совпадает с парадигмой прилагательных, но у причастий нет форм сравнительной степени. Свободные ГП для полных форм причастий – одушевленность, род, число, падеж. Свободные ГП для кратких форм причастий – род, число. ГП деепричастий – возвратность, вид, время. Свободные ГП деепричастий – время. Иногда удобно связать с глагольной лексемой чисто синтаксическую характеристику – переходность.

Личные местоимения. ГП – одушевленность, род, число, падеж, лицо. Свободная ГП личных местоимений – падеж.

Числительные. ГП – падеж и/или род. У порядковых числительных так же число.

Классы неизменяемых лексем: *Наречия; Предлоги; Союзы; Частицы; Междометия; Предикативы; Вводные слова*.

В морфологической модели выделены синтаксические подклассы лексем, имеющие определенные морфологические и/или синтаксические особенности. Например, в классе прилагательных можно выделить местоименные прилагательные (*«который»*), притяжательные прилагательные (*«дядин»*), порядковые числительные («второй») [2]. Результат морфологического анализа представлен на рисунке 3.

В модуле морфологического анализа используются следующие разработанные классы (рис. 4).

Работа модуля производится следующим образом: сначала анализируемое предложение поступает в класс Graphematic, где выполняется его графематический анализ, затем в классе MorphAnalys происходит морфологический анализ полученных в результате слов. На выходе класса MorphAnalys мы получаем список объектов типа Word, содержащих информацию о каждом слове.

Класс MorphAnalysis для своей работы использует морфологический словарь Диалинг для русского языка. Он состоит из двух файлов: morphs.mrd и rgramtab.tab. Большинство методов класса предназначены для работы с этими файлами.

СИНТАКСИЧЕСКИЙ АНАЛИЗАТОР

Для реализации синтаксического компонента лингвистического парсера было принято решение об использовании *модели отношений между словами (*метод dependency parsing*)*. Так как этот метод больше подходит для построения деревьев зависимостей для языков со свободным порядком слов, которым является русский язык.

В модуле синтаксического анализа используются классы: Tree<T> и SyntaxAnalysis (рис. 5).

Класс Tree<T> содержит поля, определяющие структуру дерева зависимостей, а также методы обращения к этим полям. Класс SyntaxAnalysis производит синтаксический анализ, для чего он обращается к базе знаний на языке Prolog. Передаваемые морфологические признаки он переводит на латиницу. Это связано с тем, что интерфейс взаимодействия C# и SWI-Prolog не предусматривает корректную обработку кириллических символов.

Для выводов результатов синтаксического анализа строится отдельная форма. Ее вид представлен на рисунке 6. Построение дерева зави-

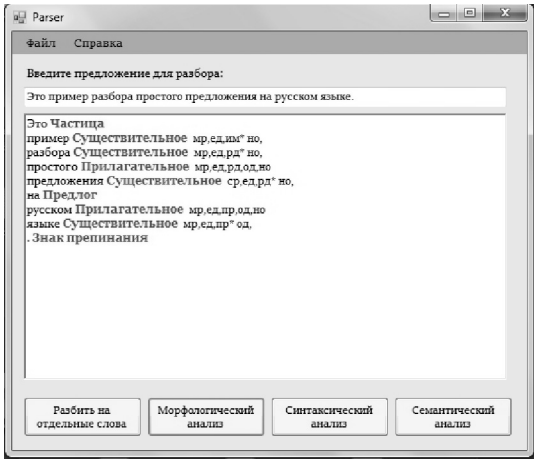

Рис. 3. Результат морфологического анализа предложения.

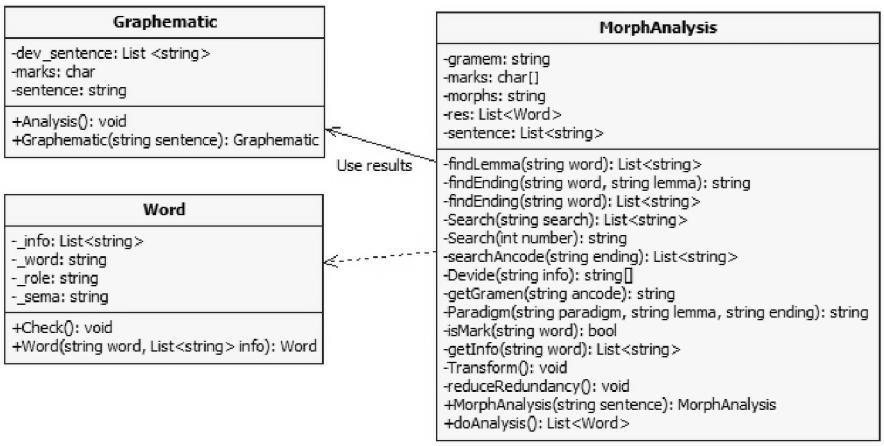

Рис. 4. Диаграмма классов модуля морфологического анализа

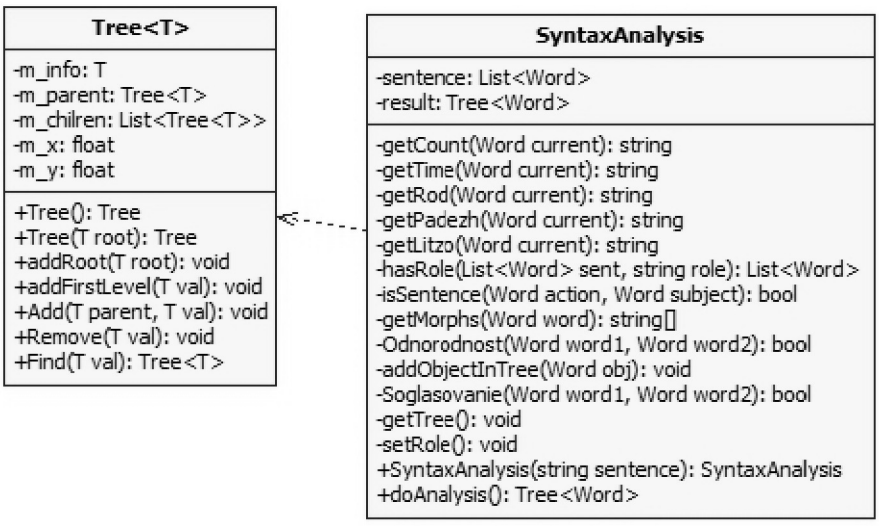

Рис. 5. Диаграмма классов модуля синтаксического анализа

симостей производится с помощью отдельного пользовательского компонента.

СЕМАНТИЧЕСКИЙ КОМПОНЕНТ

Семантический компонент производит семантический анализ текста и базируется на результатах синтаксического анализа, получая на входе уже не набор слов, разбитых на предложения, а набор деревьев, отражающих синтаксическую структуру каждого предложения.

Модуль семантического анализа состоит из двух классов: DictionaryOp и SemanticAnalysis (рис. 7). DictionaryOp является статическим классом, отвечающий за взаимодействие со словарем семантического анализа. SemanticAnalysis производит анализ поступившего на вход предложения, после чего на выход он возвращает семантическую сеть.

Семантический словарь содержит записи двух типов. Запись первого типа определяет категорию, например:

*CategoryTime.

Символ «*» в этом примере указывает на начало записи о категории, «Category» определяет тип категории, а «Time» – название категории. Типов категорий определено четыре: Category – категориям этого типа ставится в соответствие непосредственно слово или его лемма; RoleCategory – принадлежность к этим категориям определяется по роли слова в предложении; MorphCategory – уточняющие категории, соответствуют определенному морфологическому признаку; PartCategory – соответствуют определенным частям речи. Иногда после категории может в скобках указываться некий дополнительный признак, например запись вида «*MorphCategoryActor(Subject)» означает, что к категории Actor может относиться только подлежащее.

Запись второго вида – это собственно словарная статья. В первой ее части могут быть записаны слово или лемма слова, роль слова в предложении, некоторый морфологический признак или сочетание морфологических при-

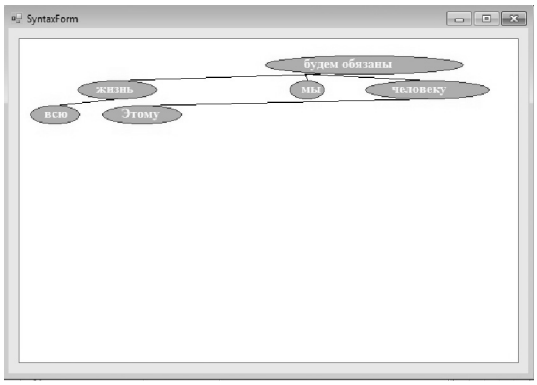

Рис. 6. Результат синтаксического анализа

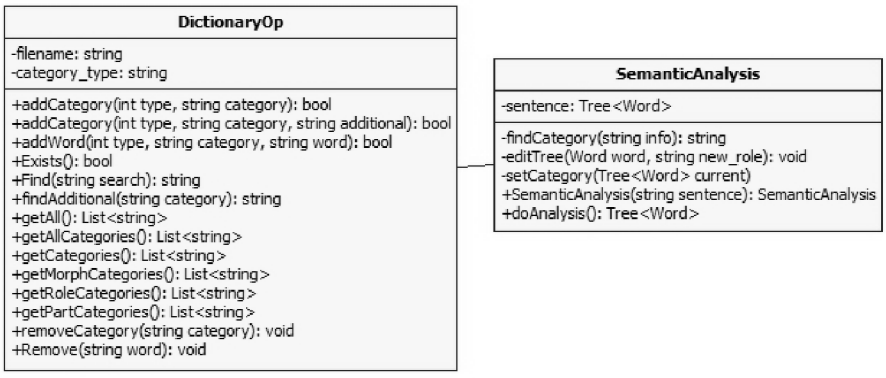

Рис. 7. Диаграмма классов модуля семантического анализа

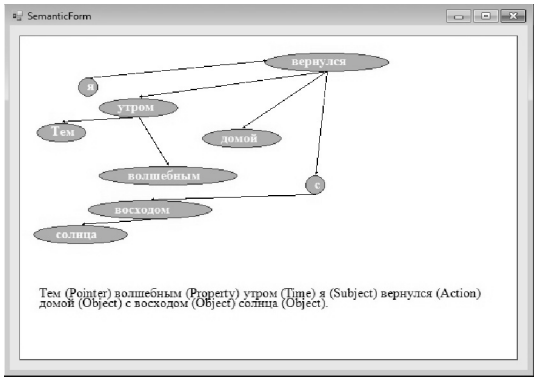

Рис. 8. Результат семантического анализа

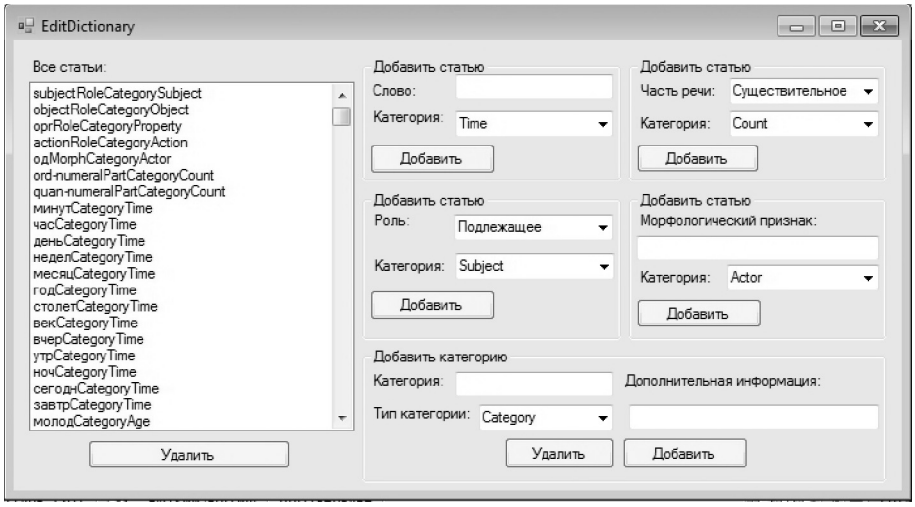

Рис. 9. Форма для редактирования семантического словаря

знаков, часть речи. Вторая часть, разделительная, содержит тип категории, третья – ее название. Пример такой записи выглядит следующим образом:

oprRoleCategoryProperty.

Это означает, что к категории Property (Свойство) могут относиться слова, которые играют в предложении роль определений.

Методы класса DictionaryOp позволяют не только взаимодействовать со словарем в рамках задачи семантического анализа, но и производить редактирование словаря.

Построение семантического дерева и определение категорий для слов производится с помощью класса SemanticAnalysis.

Результаты семантического анализа так же представлены на отдельной форме, вид которой представлен на рисунке 8.

Для редактирования семантического словаря по нажатию кнопки меню «Редактировать словарь» также создается отдельная форма. Ее вид представлен на рисунке 9.

ЗАКЛЮЧЕНИЕ

В процессе разработки парсера русского языка реализованы модули: морфологического, синтаксического, семантического анализа. Проведено тестирование, в результате он показал устойчивую работу на множестве простых предложений и частично сложных. Пока не решена проблема снятия графической омонимии. Планируется также расширить его функциональность, добавив возможность устойчивого разбора сложных предложений. Не считая этого ограничения, пользователю предоставлены все заявленные возможности, включая и возможность редактирования семантического словаря.

СПИСОК ЛИТЕРАТУРЫ

1. *Большакова Е. И., Клышинский Э. С., Ландэ Д. В., Носков А. А., Пескова О. В., Ягунова Е. В.* Автоматическая обработка текстов на естественном языке и компьютерная лингвистика : учеб. пособие // М.:МИЭМ, 2011. – 272 с. (http://clshool.miem.edu.ru/uploads/swfupload/files/011a69a6f0c 3a9c62 9d6d375f12aa27e349cb67.pdf)

2. *Волкова И. А.* Лингвистический процессор естественного языка. Морфологический и синтаксический компоненты. Задание практикума для студентов 3-го курса ЧФ МГУ (Методическое пособие) // Издательский отдел факультета вычислительной математики и кибернетики МГУ им. М. В. Ломоносова, 2002.

3. *Апресян Ю. Д., Богуславский И. М., Иомдин Л. Л.* Лингвистический процессор для сложных информационных систем. – М. : Наука, 1992, – 256 с.

4. *Зализняк А. А.* Грамматический словарь русского языка:словоизменение. – 3-е изд. М. : Рус. Яз., 1987.

5. Система «Диалинг» (http://aot.ru/index.html)

6. Парсер грамматики связей (Russian Link Grammar – http://slashzone.ru/parser/)

7. Лингвистический процессор ЭТАП-3 (http://www.cl.iitp.ru/etap3)

8. Синтаксический анализатор компании Dictim (http://www.dictum.ru/ru/syntax/blog)

9. Обзор PROMT Syntactic and Semantic Analyzer (http://www.promt.ru/company/technology/analyzer/)

10. Морфологический словарь системы «Диалинг» (http://seman.svn.sourceforge.net/viewvc/seman/branches/rustomita/Dicts/Morph/)

Гаршина Вероника Викторовна – кандидат технических наук, доцент кафедры программирования и информационных технологий факультета компьютерных наук. Воронежский государственный университет. Тел.: (473) 2208-470. E-mail: garshina@cs.vsu.ru

Богоявленская Юлия Александровна – магистр 1 года обучения, факультета Компьютерных наук, Воронежский государственный университет. Тел.: (473) 2208-470. E-mail: littlekinglet@gmail.com

Garshina Veronika V. – Candidat of Technical Sciences, Associate Professor, the dept. of Programming and Information Technologies, Voronezh State University. Tel.: (473) 2208-470. E-mail: garshina@cs.vsu.ru

Bogoyavlenskay J. A. – Master of one year of study, the dept. of Department of Computer Science, Voronezh State University. Tel.: (473) 2208-470. E-mail: littlekinglet@gmail.com