УНИВЕРСИТЕТ ИТМО
НАУЧНО-ТЕХНИЧЕСКИЙ ВЕСТНИК ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, МЕХАНИКИ И ОПТИКИ
сентябрь–октябрь 2015 Том 15 № 5 ISSN 2226-1494 http://ntv.ifmo.ru/
SCIENTIFIC AND TECHNICAL JOURNAL OF INFORMATION TECHNOLOGIES, MECHANICS AND OPTICS
September–October 2015 Vol. 15 No 5 ISSN 2226-1494 http://ntv.ifmo.ru/en

УДК 004.912:303.7

СЕМАНТИКО-СИНТАКСИЧЕСКИЙ ПАРСЕР SEMSIN

К.К. Боярский[a], Е.А. Каневский[b]

[a] Университет ИТМО, Санкт-Петербург, 197101, Российская Федерация
[b] Санкт-Петербургский экономико-математический институт РАН, Санкт-Петербург, 190013, Российская Федерация
Адрес для переписки: Boyarin9@yandex.ru

Информация о статье
Поступила в редакцию 02.06.15, принята к печати 01.07.15
doi:10.17586/2226-1494-2015-15-5-869-876
Язык статьи – русский

Ссылка для цитирования: Боярский К.К., Каневский Е.А. Семантико-синтаксический парсер SemSin // Научно-технический вестник информационных технологий, механики и оптики. 2015. Т. 15. № 5. С. 869–876.

Аннотация
Описан принцип работы семантико-синтаксического парсера SemSin, строящего дерево зависимостей для предложений русского языка. Парсер состоит из четырех блоков: словаря, морфологического анализатора, продукционных правил и лексического анализатора. Важной логической частью парсера является предсинтаксический модуль, который согласует и дополняет результаты разбора морфологического блока, разделяет абзацы текста на отдельные предложения, а также осуществляет предварительное снятие омонимии. Особенностью представляемого парсера является открытый тип управления – оно осуществляется с помощью набора продукционных правил. Богатый набор команд обеспечивает возможность как морфологического, так и семантико-синтаксического анализа предложения. Приведена последовательность применения правил, рассмотрены примеры их работы. Особенностью правил является принятие решений об установлении синтаксических связей с одновременным снятием морфологической и семантической омонимии. Лексический анализатор обеспечивает выполнение команд и правил, а также осуществляет управление парсером в ручном или автоматическом режимах разбора текста. В первом случае анализ производится интерактивно с возможностью пошагового исполнения правил и просмотра получившегося дерева разбора. Во втором случае результаты разбора записываются в xml-файл. Активное использование синтаксической и семантической словарной информации позволяет значительно уменьшить неоднозначность разбора. Кроме разметки текста, парсер может использоваться также как инструмент для извлечения информации из текстов на естественном языке.

Ключевые слова
автоматический анализ текста, актанты, дерево зависимостей, семантические классы, лексема, парсер, продукционные правила, семантика.

SEMSIN SEMANTIC AND SYNTACTIC PARSER

K.K. Boyarsky[a], E.A. Kanevsky[b]

[a] ITMO University, Saint Petersburg, 197101, Russian Federation
[b] Saint Petersburg Institute for Economics and Mathematics, RAS, Saint Petersburg, 190013, Russian Federation
Corresponding author: Boyarin9@yandex.ru

Article info
Received 02.06.15, accepted 01.07.15
doi:10.17586/2226-1494-2015-15-5-869-876
Article in Russian

For citation: Boyarsky K.K., Kanevsky E.A. SemSin semantic and syntactic parser. *Scientific and Technical Journal of Information Technologies, Mechanics and Optics*, 2015, vol. 15, no. 5, pp. 869–876.

Abstract
The paper deals with the principle of operation for SemSin semantic and syntactic parser creating a dependency tree for the Russian language sentences. The parser consists of 4 blocks: a dictionary, morphological analyzer, production rules and lexical analyzer. An important logical part of the parser is pre-syntactical module, which harmonizes and complements morphological analysis results, separates the text paragraphs into individual sentences, and also carries out pre-disambiguation. Characteristic feature of the presented parser is an open type of control – it is done by means of a set of production rules. A varied set of commands provides the ability to both morphological and semantic-syntactic analysis of the sentence. The paper presents the sequence of rules usage and examples of their work. Specific feature of the rules is the decision making on establishment of syntactic links with simultaneous removal of the morphological and semantic ambiguity. The lexical analyzer provides the execution of commands and rules, and manages the parser in manual or automatic modes of the text analysis. In the first case, the analysis is performed interactively with the possibility of step-by-step execution of the rules and scanning the resulting parse tree. In the second case, analysis results are filed in an xml-file. Active usage of

syntactic and semantic dictionary information gives the possibility to reduce significantly the ambiguity of parsing. In addition to marking the text, the parser is also usable as a tool for information extraction from natural language texts.

Keywords
automatic text analysis, actants, dependence tree, semantic classes, token, parser, production rules, semantics.

Введение

Компьютерная обработка текстов на естественном языке необходима в прикладных системах, ведущих поиск и анализ информации. Этой тематике посвящено множество работ, несколько раз проводились конкурсы, оценивающие работу анализаторов текстов (парсеров) по различным параметрам [1, 2]. Однако до настоящего времени число эффективно действующих парсеров не превышает полутора десятков, каждый из них преимущественно ориентирован на решение своего круга задач (перевод, сентимент-анализ, разметка текста и др.) и имеет свои достоинства и недостатки. В связи с этим постоянно ведутся разработки новых программных продуктов.

Оценивая результаты работы нашего морфолого-лексического анализатора TextAn [3], производящего морфологический разбор с последующим снятием омонимии, мы пришли к выводу, что для снижения остаточной омонимии до уровня в несколько процентов необходимо производить и синтаксический разбор (возможно с использованием элементов семантики). Это послужило первой причиной, побудившей авторов к разработке семантико-синтаксического парсера SemSin [4].

Был и еще один побудительный мотив. Наши исследования по автоматизации процесса извлечения онтологической информации [5] встречались с трудностями, связанными со сложностью синтаксического разбора предложений русского языка. Например, тексты терминологических словарей оказались переполненными придаточными предложениями и причастными оборотами. Исходя из этого, было решено, что парсер SemSin должен строить синтаксическое дерево зависимостей, по возможности снимать лексическую неоднозначность и управляться внешними, легко модифицируемыми, правилами. Такой способ управления позволяет достаточно квалифицированному пользователю эффективно подстраивать парсер под свои потребности и под особенности разбираемого текста.

В состав парсера SemSin входят четыре блока: словарь, морфологический анализатор, продукционные правила и лексический анализатор (рисунок). На вход парсера подается текст на русском языке, который считывается абзацами. Очередной абзац подвергается морфологическому анализу с выделением отдельных токенов (слов, словосочетаний, знаков препинания, чисел и т.д.). Затем цепочка токенов обрабатывается в лексическом анализаторе с помощью системы продукционных правил, целью которых является преобразование линейной последовательности токенов в дерево зависимостей. Рассмотрим подробнее составные части парсера.

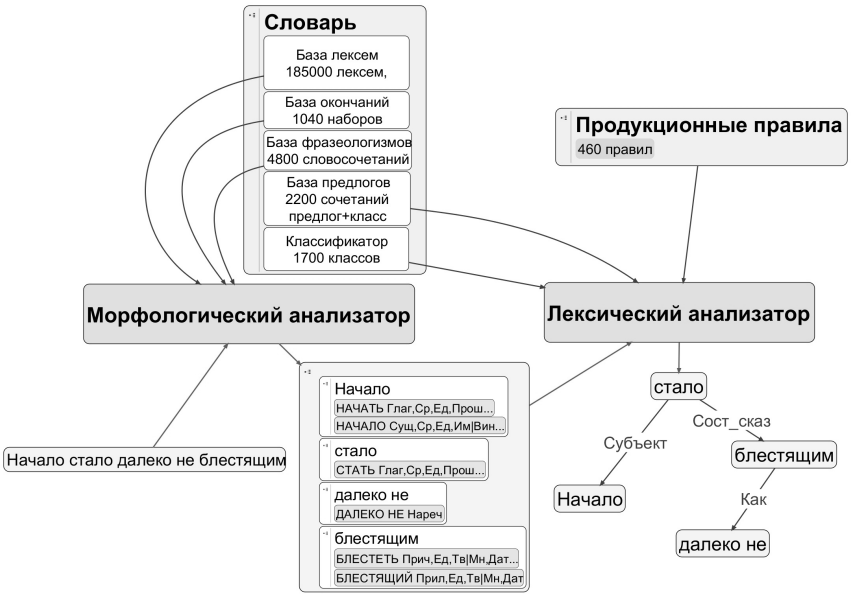

Рисунок. Схема работы парсера

Морфологический анализатор и словарь

В качестве основы использованы словарь и классификатор В.А. Тузова [6], которые за прошедшее время модифицированы и расширены. Сегодня наш словарь содержит более 190 тыс. лексем, распределенных по 1700 классам [7]. Для каждой лексемы в словаре хранятся морфологические характеристики, а также номер своего семантического класса и актанты или валентности (для подключения зависимых

слов) в виде падежей (!Им, !Род, !Вин и т.д.) или предлогов с соответствующими падежами (!вВин, !наПред и т.д.). Часто перед таким актантом указаны допустимые классы слов, могущих их замещать. Около 14% слов в словаре имеют две и более лексемы. Это могут быть слова с разными морфологическими характеристиками (например, *дворник* – человек и *дворник* – устройство, или совпадающие по написанию мужские и женские фамилии), а могут быть и слова с одинаковой морфологией, но относящиеся к разным классам (например, слову *ключ* соответствуют три лексемы, означающие инструмент, код и родник).

В системе SemSin словарь реализован в виде таблиц Excel. Морфологическая таблица содержит свыше 170 тыс. строк, некоторые из которых соответствуют не одной, а нескольким лексемам с одинаковой морфологией. Эта таблица используется морфологическим анализатором.

В парсере достаточно эффективно решается проблема устойчивых словосочетаний. Для этого служит специальная таблица фразеологизмов, которая обеспечивает разбор трех типов словосочетаний: неизменяемых (*в отличие от*, *а именно*, *в ту пору*), с изменяемым первым словом (*дым коромыслом*) и полностью изменяемых (*синий чулок*). В настоящее время эта таблица содержит более 4800 фразеологизмов и играет важную роль в снятии неоднозначности, особенно для составных предлогов, союзов и наречий.

Важным элементом словарного обеспечения является таблица предлогов, хранящая более 2200 сочетаний классов существительных, с которыми взаимодействуют предлоги, и названия связей с хозяином, к которому присоединяется предложная группа.

Морфологический анализатор [8, 9] осуществляет разбор очередного слова, поданного на его вход. Результат разбора выдается в виде леммы (слова в нормальном виде) с морфологическими характеристиками, а также класса с указанием соответствующих актантов. Морфологический анализатор реализован в виде DLL-файла на C++.

Предсинтаксический модуль

Известно [10], что предсинтаксический модуль облегчает проведение синтаксического разбора. В нашей системе этот модуль [11] служит промежуточным звеном между морфологическим анализатором и остальной частью парсера, реализован как часть лексического анализатора и выполняет несколько функций.

Прежде всего, убираются крайне редко встречающиеся словоформы деепричастий и глаголов в повелительной форме (*весь*, *всей*, *для*, *зря*, *мая*, *моря и т.д.*).

В ряде случаев требуется уточнение актантов:

- для переходных глаголов с отрицанием к актанту «!Вин» добавляется актант «!Род» с теми же классами (*купить книгу – не купить книги*);
- для непереходных глаголов к списку актантов добавляется «!Род» (*я не набирался опыта*);
- для страдательных причастий актант «!Им» заменяется «!Тв» с теми же классами, что обеспечивает согласование ролей (*Мама испекла пирог – Пирог*, *испеченный мамой*).

Затем производится токенизация, т.е. выделение минимальных линейных компонент текста, которые в дальнейшем рассматриваются как неделимые единицы.

Определенную трудность при токенизации представляют слова с дефисами [12]. Сейчас в словаре таких слов около 4,5 тыс. Однако словотворчество в данной области настолько распространено, что все варианты охватить невозможно. Самый простой вариант, если к имеющемуся в словаре слову добавляется стандартная частица из списка «*-то*, *-ка*, *-де*, *-ко*, *-та*, *-те*, *-с*, *-либо*, *-нибудь*, *-таки*». В этом случае при морфологическом анализе частица просто отбрасывается, определяются грамматические характеристики основного слова, а затем присоединяется частица.

В более сложных случаях морфоанализатор обрабатывает такое слово по частям, иногда выдавая несколько лемм на каждую часть. Например, в слове *серо-зеленый* первая часть имеет первоначально два варианта разбора (наречие и краткое прилагательное) и вторая часть тоже имеет два варианта (прилагательное и существительное). Собирается сложная лемма, к которой приписываются грамматические характеристики второй части (в данном примере однозначно получается прилагательное).

Грамматические характеристики второй части являются определяющими, если первая часть неизвестного слова с дефисом написана латинскими буквами (чаще всего на английском языке). Такими словами могут быть *internet-кафе*, *ip-сеть*, *web-браузер* и т.п. В этом случае первая часть слова запоминается, а вторая обрабатывается морфоанализатором, после чего к полученному результату приписывается первая часть.

Особым образом обрабатываются словосочетания цифра – цифра (*2–3 года*) или цифра – буквенное окончание длиной до трех букв (*3-ему периоду*). Для анализа последних используется специальная таблица окончаний, позволяющая определить часть речи, возможные род и падеж.

Как бы обширен ни был морфологический словарь, в тексте обязательно найдутся неизвестные слова. В системе SemSin сделана попытка спрогнозировать синтаксические и семантические значения таких слов. В ряде случаев неизвестное слово может представлять собой сочетание двух известных слов, написанных без дефиса (*шестиметровый*), или известного слова со стандартной приставкой (*архиплут*).

Для анализа таких словосочетаний используются некоторые стандартные основы слов (*метров*, *дюймов*, *тонн*, *часов* и др.) или наиболее распространенные приставки (*авиа-*, *агро-*, *анти-*, *архи-*, *баро-*, *видео-*, *внутри-* и др. [13]). Аналогичные методы применяются и в нашем парсере.

Для неизвестных слов, начинающихся с прописной буквы, производится анализ их окружения. Так, если слева от неизвестного слова X расположен токен, состоящий из прописной буквы с точкой или частицы типа *аль*, *аф*, *бен*, *ван* и др., тогда это слово – фамилия. Если же слева стоит слово из особого «географического» списка (*долина*, *залив*, *звезда*), то X – название объекта. Если слово X заключено в кавычки, а слева имеется токен, обозначающий учреждение или предприятие, то предполагается, что это слово – название предприятия (*завод «Синтаг»*).

Если неизвестное слово заканчивается на определенный набор окончаний, то делается попытка рассматривать его как русскоязычную фамилию [14]. Естественно, что прежде всего нас заинтересовали фамилии с наиболее часто встречающимися окончаниями (в частности, *-ов*, *-ев*, *-ин* и суффиксом *–ск-*) в мужском и женском вариантах. Отметим, что если основа оканчивается на -ск, то это слово может быть не только фамилией (*Бур-Комаровский*), но и прилагательным, определяющим название чего-либо (*Новокраматорский завод*).

Одним из путей уменьшения омонимии является широкое использование стандартных сочетаний слов – фразеологизмов [10]. Их можно разделить на три группы: неизменяемые, с изменяемым первым словом и полностью изменяемые.

При синтаксическом разборе предложения в парсере SemSin неизменяемый фразеологизм рассматривается как один токен. Слова, входящие в состав изменяемых фразеологизмов (*выйти замуж*, *дым коромыслом*, *антонов огонь*, *синий чулок*), в единый токен не объединяются, но сразу соединяются в лексическую группу подходящей связкой. При этом группе приписывается соответствующий семантический класс. Так, *дым коромыслом* не будет идти из трубы, а *синий чулок* получит класс людей.

Неизменяемые фразеологизмы, по существу, являются неоднословными лексическими оборотами, эквивалентными слову. Они могут быть предлогами (*в зависимости от*, *несмотря на*), наречиями (*в конце концов*, *время от времени*), союзами (*а также*, *если бы*), частицами (*все же*, *вроде бы*), вводными оборотами (*к слову сказать*, *другими словами*) и предикативными оборотами (*не дай бог*, *лыка не вяжет*). Достаточно полные их списки приведены в Национальном корпусе русского языка (НКРЯ) [15] и у Р.П. Рогожниковой [16].

В парсере SemSin снятие грамматической и частеречной омонимии производится одновременно с построением синтаксического дерева зависимостей. Однако, когда попадаются словоформы, обладающие высокой степенью омонимии, возникает необходимость снять или хотя бы уменьшить эту омонимию до начала синтаксического разбора. В ряде случаев это достигается путем анализа ближайшего контекста слева и справа [17]. Более сложный тип омонимии снимается путем анализа и более удаленного контекста, если соответствующие слова обладают четко выраженными семантическими характеристиками. Иногда снять омонимию удается и за счет графематики, в частности, путем анализа наличия заглавных букв в самом омонимичном слове.

Правила

После завершения работы предсинтаксического модуля цепочка токенов обрабатывается с помощью системы продукционных правил [18]. Каждое из правил, общее число которых превышает 400, применяется последовательно ко всем токенам. При этом время анализа линейно связано с числом поданных на разбор слов и практически не зависит от длины отдельного предложения.

Разработанный нами язык записи правил достаточно адекватен как решаемой задаче, так и условию простоты отладки. Правило состоит из нескольких частей, для каждой из которых используются свои операторы.

1. Имя правила – уникальный идентификатор, позволяющий обращаться к данному правилу из других правил. Правила, имена которых начинаются с «SR» (SlaveRule), сами по себе не исполняются, они только могут быть вызваны из других правил (MasterRule) и служат в основном для анализа левого и правого контекста обрабатываемого слова. Если имя начинается с символов «RToL», то данное правило применяется к токенам справа налево. Остальные правила применяется к токенам в обычном порядке (слева направо). Заметим сразу, что в процессе разбора предложение делится на сегменты согласно [19].

Каждая переменная принимает значение (адрес), соответствующее позиции токена в цепочке, т.е. позицию относительно начала абзаца. Имена переменных начинаются с символа #. Первая переменная всегда указывает на обрабатываемый в данный момент токен. В правилах типа Slave значение первой переменной задается в точке вызова. Для задания остальных переменных используются команды, устанавливающие адрес относительно другой переменной, положения в сегменте или наличия заданной связи.

В связи с тем, что по мере срабатывания правил линейность структуры токенов нарушается за счет появления блоков (сегменты, именные и предложные группы и т.д.), нам пришлось ввести три типа

переменных. Если имя переменной начинается с *#W*, то при попадании внутрь именной группы переменная сдвигается на ее вершину. Это позволяет пропускать зависимые лексемы (например, части составных числительных, прилагательные и т.п.) при дальнейшем анализе предложения. Если имя переменной начинается с *#Z*, то такого сдвига не происходит. Эти переменные локализованы в пределах сегмента, а переменные типа *#Y* позволяют выходить за его границу.

2. Условная часть правил строится по обычной для языков программирования схеме *If…Then…ElseIf…Then…Else…EndIf*. Внутри каждого блока могут использоваться операторы конъюнкции *&* и дизъюнкции *OR*. Разрешено использовать вложенные операторы *If*.

Первая группа операторов определяет позицию токена в абзаце или предложении. Вторая группа операторов проверяет тип токена (его графематику). Третья группа анализирует морфологические характеристики слова и его класс. Четвертая проверяет согласование слов по роду, числу и падежу, а также согласование аргументов подсоединяющего слова с падежами и классами подсоединяемого слова. Пятая группа операторов анализирует фрагменты уже созданного синтаксического дерева. Всего в условной части правил использовано более 70 типов операторов проверки.

3. Если все проверявшиеся условия удовлетворены, начинают выполняться команды исполнительной части. Эти команды устанавливают связи между токенами (в том числе и кореферентные), уменьшают неоднозначность разбора за счет удаления морфологических характеристик или целых лексем, позволяют изменять морфологические и семантические характеристики токенов, обеспечивают сегментацию предложений. Также обеспечивается согласование слов (например, по роду, числу и падежу – прилагательных и причастий с существительными, по аргументам существительных – с другими существительными и предикатами и т.д.). Правила могут запускаться или безусловно, или в зависимости от результатов работы предыдущего правила. Всего в исполнительной части использовано около 100 команд.

4. Рассмотрим логику работы правил на примере дифференциации существительного и наречия по словоформе *потом* на примерах, взятых из НКРЯ. Для этого используются четыре правила: MasterRule <ПОТОМ>, SlaveRule <SR:ПОТОМ1–> и <SR:ПОТОМ1а–>, проверяющие левый контекст, и <SR:ПОТОМ2+>, проверяющее правый контекст. Во всех примерах правило <ПОТОМ> обнаруживает соответствующую словоформу и вызывает подчиненные правила (SlaveRule).

4.1. *Кусок земного металла*, *жаркий слиток земных надежд*, *продукция мозга и мышц*, *смешанная с нашим потом и с кровью тех*, *которые этого уже не услышат.*

Правило <SR:ПОТОМ1а–> осуществляет поиск предлога «*с*/*со*», последовательно проверяя левый контекст вплоть до знака препинания, глагола во всех формах или до начала предложения, и оставляет существительное *пот*.

4.2. *Зачем мы обливаемся потом и падаем на каждом шагу от усталости.*

Правило <SR:ПОТОМ1а–> предлога не обнаруживает. Вызывается правило <SR:ПОТОМ1–>, которое находит слева глагол *обливаться*, способный подсоединить существительное нужного класса. Оставляет существительное *пот*.

4.3. *Милия Алексеевича едва потом не прошибло.*

Правила <SR:ПОТОМ1а–> и <SR:ПОТОМ1–> не срабатывают, вызывается правило <SR:ПОТОМ2+>, которое находит справа подходящий глагол (*прошибить*). Оставляет существительное *пот*.

4.4. *Сдадите ли потом квартиру или просто комнату*.

Правила <SR:ПОТОМ1а–>,<SR:ПОТОМ1–> и <SR:ПОТОМ2+> не срабатывают (глагол *сдавать* не имеет подходящего актанта), поэтому остается наречие *потом*.

Лексический анализатор

Лексический анализатор является ядром всей системы. Результаты морфологического разбора заносятся в линейную структуру PipeLine в порядке, соответствующем порядку токенов в предложении, причем в каждом элементе может размещаться до семи лексем с разными морфологическими характеристиками. Затем система применяет каждое из правил типа MasterRule последовательно ко всем токенам.

В каждом элементе структуры содержится полный набор морфологических характеристик: лемма, часть речи, тип склонения, род, лицо, падежи единственного и множественного числа, признак кратких прилагательных и причастий, классы лексем и их актанты. Для глаголов дополнительно хранятся род/число или лицо/число, время, а также форма глагола (инфинитив, деепричастие, причастие) и залог для причастия. Местоименные прилагательные и порядковые числительные обрабатываются как прилагательные.

Кроме того, каждый элемент хранит дополнительные характеристики слова, в том числе количество лексем, тип знака препинания, регистр букв, тип входящей связи, тип именной группы и др. Для работы с сегментами имеется информация о центре сегмента, его типе (вводный, причастный/деепричастный оборот и т.д.) и границах.

Лексический анализатор реализован на Visual Basic и, прежде всего, обеспечивает считывание исходного текста. Обрабатываются результаты морфологического анализа и разбираются фразеологизмы. Затем выполняются правила. Для удобства составления и контроля последовательности выполнения правил они делятся на группы, каждая из которых представляет собой обычный текстовый файл. Отметим, что совокупность правил не может быть представлена в виде простого набора конъюнкций условий, поскольку порядок их выполнения критичен для получения правильного разбора. В системе принята следующая последовательность исполнения правил (три первые файла являются составной частью предсинтаксического модуля).

1. Объединение сложных единиц и Интернет-адресов в одно слово.
2. Разбиение абзаца на предложения. Выявление инициалов и аббревиатур [20].
3. Снятие морфологической и семантической омонимии для отдельных слов.
4. Разбор имен собственных типа ФИО и названий (*оз. Байкал* и *капитан Иванов*).
5. Анализ чисел (физических величин, дат и адресов) [21].
6. Разбор вводных слов. Разрешение коллизий существительное–прилагательное. Обработка числительных.
7. Снятие неоднозначностей у прилагательных и причастий.
8. Обработка предлогов и образование предложных групп.
9. Выделение причастных и деепричастных оборотов, придаточных предложений. Удаление глагола в деепричастных оборотах. Подключение наречий оценки и времени.
10. Выделение сказуемых и однородных членов предложения с союзом «и».
11. Анализ переходных глаголов и выделение прямых дополнений.
12. Выделение однородных сказуемых, связанных запятой. Объединение сегментов. Анализ сочетания глагола *быть* с родительным падежом.
13. Подключение существительных в родительном падеже к «хозяину».
14. Выделение составных сказуемых, анализ инфинитивов и подключение к ним прямых дополнений. Определение подлежащих.
15. Подключение предложных групп к «хозяину». Подключение оставшихся наречий.
16. Соединение сегментов. Подключение вводных оборотов и придаточных предложений. Выявление сложносочиненных предложений.
17. Подключение тире и словосочетания *это есть*. Окончательная обработка дерева связей: подключение предлога и союза к центру предложения (глаголу), повторное согласование фамилии, имени и отчества, а также прилагательного с существительным.
18. Объединение предложений в абзаце.
19. Выявление анафорических отношений для личных, притяжательных, возвратных местоимений, а также местоимений *который* и *этот* [22].

Кроме файлов правил, в парсере используется еще три дополнительных файла.

– Файл «Collect» используется для хранения групп слов, имеющих общие характеристики, например, аббревиатуры, названия географических объектов и т.д.
– Файл «fNotCor» содержит нестандартные словоформы (устаревшие, просторечные), которые при анализе заменяются стандартными, например, *жисть* – на *жизнь*.
– Файл «UnUse1» служит для настройки на предметную область. Перечисленные в нем лексемы с классами исключаются из анализа. Например, для финансовой области исключение слова БАНКА $121311114 позволяет избежать сложностей с разрешением омонимии с формами слова БАНК.

Для удобства обеспечивается два варианта отладки правил. Можно выполнять последовательность правил в пошаговом режиме или выбрать интересующее пользователя правило и выполнять его по одному оператору.

Результатом работы парсера в ручном режиме является отображение полученного дерева разбора, а в автоматическом – xml-файл. В этом файле содержатся исходный текст предложения, грамматические описатели и семантический класс каждой словоформы, а также тип входящей связи и ее источник.

Заключение

Еще раз подчеркнем, что основным результатом работы анализатора SemSin мы считаем правильно построенное дерево зависимостей с однозначно определенными морфологическими параметрами узлов. Отметим, что за счет включения кореферентных связей в этом дереве появляются элементы сети.

На основе результатов анализа возможна дальнейшая обработка, проводимая с различными целями: семантическая разметка, построение онтологий и сценариев и т.д. [23–25]. Поскольку в этом смысле результаты SemSin носят промежуточный характер, мы не стремились к унификации названий связей с какой-либо из известных систем анализа, тем более что стандарта де-факто в этой области не существует. Большинство связей в нашей системе именуются синтаксически (*я* (Субъект) ←*люблю*), либо по падежам

подключаемого слова (*прочитал* → (Вин) *газету*), либо по сочетанию предлог-падеж (*живу* → (наВин) *на средства*), либо в соответствии с семантикой (*живу* → (Где) *на море*).

Проведенные исследования показали, что семантико-синтаксический парсер, построенный на основе применения продукционных правил, позволяет достаточно успешно строить деревья подчинения. По результатам работы парсера неоднозначно определяются леммы примерно в 0,7% случаев (например, *охарактеризовал его как человека…* – по словоформе *его* остаются леммы ОН/ОНО). Остаточная морфологическая неоднозначность, т.е. род, число, время и т.д. – порядка 3–4% (например, в предложении *Разрешите к вам присоединиться*? – у глагола *разрешить* остается повел. наклонение / будущее время).

References

1. Lyashevskaya O.N., Astaf'eva I., Bonch-Osmolovskaya A., Gareishina A., Grishina Yu., D'yachkov V., Ionov M., Koroleva A., Kudrinskii M., Lityagina A., Luchina E., Sidorova E., Toldova S., Savchuk S., Koval' S. Evaluation methods for automatic text analysis: morphological parsers of Russian language. *Computational Linguistics and Intelligent Technologies,* 2010, no. 9 (16), pp. 318–326. (In Russian)
2. Toldova S.Yu., Sokolova E.G., Astaf'eva I., Gareishina A., Koroleva A., Privoznov D., Sidorova E., Tupikina L., Lyashevskaya O.N. Evaluation methods for automatic text analysis 2011-2012: syntax parsers of Russian language. *Computational Linguistics and Intelligent Technologies,* 2012, no. 11, pp. 77–90. (In Russian)
3. Kanevsky E.A., Boiarsky K.K. Morphological and lexical analyzer and text classification. *Materialy V Mezhdunarodnoi Nauchno-Prakticheskoi Konferentsii Prikladnaya Lingvistika v Nauke i Obrazovanii* [Proc. V Int. Scientific Conference on Applied Linguistics in Science and Education]. St. Petersburg, 2010, pp. 157–163. (In Russian)
4. Kanevsky E.A., Boiarsky K.K. The semantic-and-syntactic parser SemSin. *Computational Linguistics and Intelligent Technologies*. 2012.
5. Boyarsky K.K., Kanevsky E.A., Lezin G.V., Kalinichenko L.A., Skvortsov N.A. Automation of process of extraction of the ontological information from verbal terminological dictionaries (on the example of the terminological dictionary of the problem of interstellar extinction). *Proc. XII Conference on Digital Libraries: Advanced Methods and Technologies, RCDL-2010*. Kazan', 2010, pp. 257–264. (In Russian)
6. Tuzov V.A. *Komp'yuternaya Semantika Russkogo Yazyka* [Computer Semantics of Russian Language]. St. Petersburg, SPbSU Publ., 2004, 400 p.
7. Boyarsky K.K., Kanevsky E.A., Stafeev S.K. The use of dictionary information in text analysis. *Scientific and Technical Journal of Information Technologies, Mechanics and Optics*, 2012, no. 3(79), pp. 87–91.
8. Kanevsky E.A., Kolpakova N.V. On the construction of the morphological analyzer. *Computational Linguistics and Intelligent Technologies,* 1999, vol. 2, pp. 98–106. (In Russian)
9. Boyarskii K.K., Kanevskii E.A., Klimenko E.N. Morphological text analysis in MAZE-32. *Informatsionnye Tekhnologii v Gumanitarnykh i Obshchestvennykh Naukakh*. St. Petersburg, SPb EMI RAN, 2001, no. 11, pp. 1–8. (In Russian)
10. Kobzareva T.Yu., Afanas'ev R.N. Universal pre-syntactical module of homonymy parts of speech in Russian using dictionary-based diagnostic situations. *Computational Linguistics and Intelligent Technologies*, 2002, pp. 258–268. (In Russian)
11. Boyarsky K.K., Kanevsky E.A. Pre-syntactical module of the parser SemSin. *Internet i Sovremennoe Obshchestvo*. St. Petersburg, 2013, pp. 280–286.
12. Dorokhina G.V., Zhuravlev A.O., Bondarenko E.A. Study algorithm of morphological analysis of words with spelling defisnym. *Sistemy i Sredstva Iskusstvennogo Intellekta, SSII-2012*. Donetsk, 2012, pp. 17–24. (In Russian)
13. Zakharov V.P. Morphological analysis of unfamiliar words in the text based on word-formation models. *Materialy XLIV Mezhdunarodnoi Filologicheskoi Konferentsii* [Proc. XLIV International Philological Conference]. St. Petersburg, 2015, pp. 581–582. (In Russian)
14. Boyarskii K.K., Kanevskii E.A. Automatic detection of surnames in the text. In *Informatsionnye Sistemy dlya Nauchnykh Issledovanii*. St. Petersburg, 2012, pp. 280–286. (In Russian)
15. *Natsional'nyi Korpus Russkogo Yazyka*. Available at: http://www.ruscorpora.ru/ (accessed: 2.03.2015).
16. Rogozhnikova R.P. *Tolkovyi Slovar' Sochetanii, Ekvivalentnykh Slovu* [Explanatory Dictionary of Combinations Equivalent to Word]. Moscow, Astrel'-AST Publ., 2003, 416 p.
17. Kanevsky E.A., Boyarsky K.K. Special words in the Russian language text. *Materialy XLII Mezhdunarodnoi Filologicheskoi Konferentsii* [Proc. XLII International Philological Conference]. St. Petersburg, 2013, pp. 47–52. (In Russian)
18. Boyarsky K.K., Kanevsky E.A. Rule language for construction of a syntactic tree. *Internet i Sovremennoe Obshchestvo, IMS-2011*. St. Petersburg, 2011, pp. 233–237. (In Russian)
19. Kobzareva T.Yu. Principles of segmentation analysis of Russian sentences. *Moskovskii Lingvisticheskii Zhurnal*, 2004, vol. 8, no. 1, pp. 31–80. (In Russian)

20. Boyarskii K.K., Kanevskii E.A. Splitting text into sentences. *Diskussiya Teoretikov i Praktikov*, 2010, no. 3, pp. 135–137. (In Russian)
21. Avdeeva N.A., Boyarskii K.K. About the syntactical relation in numerical constructions. *Materialy XLIV Mezhdunarodnoi Filologicheskoi Konferentsii* [Proc. XLIV International Philological Conference]. St. Petersburg, 2015, pp. 569–570. (In Russian)
22. Boyarsky K.K., Kanevsky E.A., Stepukova A.V. Anaphoric relations identification by automatic text analysis. *Scientific and Technical Journal of Information Technologies, Mechanics and Optics*, 2013, no. 5(87), pp. 108–112. (In Russian)
23. Boyarskii K.K., Kanevskii E.A., Lezin G.V. Preliminary transform of the syntax tree. *Internet i Sovremennoe Obshchestvo*. St. Petersburg, 2010, pp. 3–8. (In Russian)
24. Artemova G., Boyarsky K., Gusarova N., Dobrenko N., Kanevsky E. Text categorization for generation of historical shipbuilding ontology. *Proc. XVI Conference on Digital Libraries: Advanced Methods and Technologies, RCDL-2014*. Dubna, Russia, 2014, pp. 159–164.
25. Artemova G., Gouzévitch D., Gusarova N., Dobrenko N., Kanevsky E., Petrova D. Text categorization for generation of historical shipbuilding ontology. *Communications in Computer and Information Science*, 2014, vol. 468, pp. 1–14.

Боярский Кирилл Кириллович – кандидат физико-математических наук, доцент, доцент, Университет ИТМО, Санкт-Петербург, 197101, Российская Федерация, Boyarin9@yandex.ru

Каневский Евгений Александрович – кандидат технических наук, ведущий научный сотрудник, Санкт-Петербургский экономико-математический институт РАН, Санкт-Петербург, 190013, Российская Федерация, kanev@emi.nw.ru

Kirill K. Boyarsky – PhD, Associate Professor, Associate Professor, ITMO University, Saint Petersburg, 197101, Russian Federation, Boyarin9@yandex.ru

Evgeny A. Kanevsky – PhD, leading scientific researcher, Saint Petersburg Institute for Economics and Mathematics, RAS, Saint Petersburg, 190013, Russian Federation, kanev@emi.nw.ru