Большакова Е.И.

Язык лексико-синтаксических шаблонов LSPL: опыт использования и пути развития

Введение

Задача поиска и извлечения информации из текстов на естественном языке (ЕЯ) все чаще возникает в практике автоматической обработки текстов. В ее рамках выполняется выделение в тексте именованных сущностей (имен, персоналий, географических названий), связанных с ними событий [3, 14] – эта информация необходима в разных приложениях текстовой аналитики. К этой задаче часто относят также автоматическое выявление терминов (слов и словосочетаний предметной области), что необходимо в таких приложениях, как аннотирование тестов, создание тезаурусов и онтологий.

Решение указанных задач требует распознавания в тексте на ЕЯ определенных языковых конструкций (в частности, именных и глагольных групп), что реализуется обычно на базе поверхностного синтаксического анализа. Для упрощения разработки конкретных приложений все чаще используются инструментальные системы [2, 8, 10], обычно предлагающие формальные языки для задания информации о составе и грамматических свойствах распознаваемых конструкций, в форме специальных шаблонов. С помощью шаблонов уже готовые программные модули по анализу ЕЯ-текста настраиваются для решения требуемой задачи автоматической обработки текстов.

Наиболее известной из инструментальных систем, ориентированных на построение ЕЯ-приложений, является система GATE [2] с языком Jape для записи шаблонов [1]. В то же время Jape не содержит лингвистической специфики, что требует определенной настройки на язык анализируемых текстов. Для обработки текстов на русском языке был разработан ряд отечественных систем со своими средствами задания шаблонов, в частности, RCO Pattern Extractor [8] и система Alex [10]. Применяемые в этих инструментальных системах языки шаблонов имеют ограничения на задание лингвистической информации. Появление в последние годы новых формализмов для записи лингвистических шаблонов и поддерживающих их систем [11, 12], среди которых особо выделяется инструментальная система Томита-парсер [13], свидетельствует о том, что проблема эффективного описания лингвистических свойств ЕЯ-выражений для их автоматического выделения в тесте все еще далека от полного решения.

Рассматриваемый в данной работе язык лексико-синтаксических шаблонов LSPL [4,5] создавался как декларативный язык спецификации лексических и грамматических свойств конструкций, выделяемых в

текстах на русском языке, с целью автоматизации ряда задач обработки научно-технических текстов. Для LSPL был разработан метод автоматического выделения в тексте конструкций по их шаблонам и реализован поддерживающий программный комплекс [6], включающий среду с графическим пользовательским интерфейсом для просмотра и анализа текстов по LSPL-шаблонам. Подготовка и настройка шаблонов выполняется лингвистом или специалистом по предметной области анализируемых текстов, без участия программиста.

Язык LSPL и поддерживающие его программные средства были применены при создании нескольких приложений, требующих анализа ЕЯ-текстов, что позволило, с одной стороны, оценить выразительность языка и эффективность метода поиска по лексико-синтаксическим шаблонам, а с другой стороны – проанализировать возникающие при настройке шаблонов проблемы и наметить пути дальнейшего развития языка. Поскольку в ряде прикладных задач необходимо преобразование языковых выражений, выявленных в тексте с помощью шаблонов, язык LSPL был дополнен новыми средствами.

В данной работе кратко характеризуются и сравниваются выразительные возможности языка LSPL и языков шаблонов наиболее известных систем извлечения информации из текстов. Обсуждаются созданные с помощью LSPL приложения по обработке текстов и описываются включенные в язык новые средства, упрощающие построение ЕЯ-приложений. В заключении формулируются направления дальнейшего развития средств LSPL с целью повышения его гибкости и выразительности.

Шаблоны ЕЯ-конструкций и язык LSPL

При создании языка LSPL были проанализированы и учтены разные аспекты известных к тому моменту языков шаблонов, в первую очередь – языка Jape системы GATE [1, 2] и языков систем, ориентированных на обработку текстов на русском языке.

Язык Jape [1] представляет из себя язык правил для преобразования произвольной текстовой разметки в форме аннотаций, приписываемых к непрерывным фрагментам обрабатываемого текста. Достоинством языка является его универсальность, однако ее следствие – отсутствие встроенных лингвистических атрибутов аннотаций (лексических, морфологических, синтаксических). К недостаткам можно отнести громоздкость получающихся шаблонов языковых конструкций, а также невозможность сравнения атрибутов разных аннотаций одной и той же конструкции, что затрудняет запись условия грамматического согласования её элементов, характерного для многих конструкций русского языка (в частности, именных словосочетаний).

В отечественной системе RCO Pattern Extractor [8], построенной на принципах системы GATE, язык шаблонов был расширен средствами

задания морфологических характеристик слов выделяемых ЕЯ-выражений (на базе модулей анализа русской морфологии). Тем не менее, язык шаблонов сохраняет основные недостатки исходного языка Jape, такие как громоздкость шаблонов и сложность задания в них условия грамматического согласования.

В системе Alex [10], разработанной для целей автоматизации лингвистических исследований и подготовки текстов, использовались лексические шаблоны. В простейшем случае шаблон состоит из нескольких лексем, а в более сложных описывает несколько вариантов языковой конструкции с использованием опциональных лексем и уже определенных вспомогательных шаблонов. В целом, язык лексических шаблонов системы Alex достаточно лаконичен и декларативен, однако в нем нет средств задания морфологических характеристик слов (часть речи, падеж, число и др.), а также их грамматического согласования.

В языке шаблонов DSTL недавно разработанной системы лексического анализа ЕЯ-текстов DICTASCOPE TOKENIZER [12] реализованы некоторые новые идеи (в частности, набор встроенных функций для проверки элементов шаблона), однако в целом мощность этого языка не превосходит, например, язык системы RCO.

Язык лексико-синтаксических шаблонов LSPL [4, 5] представляет собой декларативный язык со встроенными средствами задания лексических и поверхностно-синтаксических свойств конструкций, выявляемых в русскоязычных текстах. Элементами шаблона могут быть входящие в конструкцию слова и словосочетания, а также условия их грамматического согласования. Для элементов-слов в общем случае указываются часть речи и лексема, конкретизируются морфологические характеристики (падеж, род, число и т.п.). Условия согласования задают равенство либо конкретных морфологических характеристик, либо всех общих морфологических признаков согласуемых слов. К примеру, шаблон `N V<t=pres> Av <N=V>` описывает сочетание существительного (`N`), следующего за ним глагола (`V`) в прошедшем времени (`t=past`) и наречия (`Av`), причем существительное и глагол грамматически согласованы (как в текстовом фрагменте: *модуль работает быстро*).

В LSPL-шаблонах можно задавать повторения элементов, опциональные элементы, а также альтернативные варианты языковой конструкции (для этого используются известные метасимволы `{`, `}`, `[`, `]`, `|`) – указанные возможности эквиваленты операторам регулярных выражений. Также есть возможность определять для шаблона имя и параметры, которые фиксируют некоторые морфологические характеристики описываемой конструкции. К примеру, шаблон

```
NP = {A} N1 <A=N1> [N2<c=gen>] (N1)
```

определяет именную группу `NP` из нескольких прилагательных, согласованного с ними существительного и опционального существительного в родительном падеже (*логический вывод*, *мощная реактивная сила*, *короткий интервал времени* и т.п.); параметрами этого шаблона установлены морфологические характеристики первого существительного. Уже определенные шаблоны можно применять для задания шаблонов более сложных ЕЯ-выражений, используя при этом их параметры для конкретизации или согласования элементов описываемой конструкции. Например, на основе шаблона `NP` можно описать конструкцию, состоящую из именной группы и согласованного с ней глагола в прошедшем времени: `NP V<t=past> <NP=V>` (*распад частиц фиксировался*).

Набор взаимосвязанных LSPL-шаблонов фактически задает КС-грамматику (расширенную условиями) для выявляемой в тексте языковой конструкции. Схожим образом, с помощью расширенной КС-грамматики определяется извлекаемая конструкция и в программном инструменте Томита-парсер [5] компании Яндекс, при создании которого были учтены и реализованы ключевые возможности уже существующих систем извлечения информации из текстов.

В Таблице 1 представлены основополагающие возможности формальных языков для описания выделяемых в русскоязычных текстах ЕЯ-выражений (включены языки, применяемые в рассмотренных выше системах извлечения информации).

Таблица 1.Сравнение языковых средств систем

Выразительные возможности языков шаблонов	Alex	RCO	Томита	LSPL
Задание морфологических характеристик слов	–	+	+	+
Грамматическое согласование	–	–	+	+
Вспомогательные шаблоны	+	+	+	+
Строки символов	+	+	–	+
Логические операции	+	+	+	–
Операторы регулярных выражений и их аналоги	+	+	+	+

Видно, что рассмотренные языки сопоставимы по возможностям. Все они допускают вспомогательные шаблоны (правила), а также операции регулярных выражений или их аналоги, служащие для записи альтернатив, повторяющихся и опциональных конструкций. Все системы, кроме Alex дают возможность задавать ограничения на морфологические характеристики слов. Три из четырех систем

разрешают логическое комбинирование условий: Alex – только лексических, Томита-парсер – морфологических, а RCO – как лексических, так и морфологических. Однако грамматическое согласование, необходимое для однозначного выделения определенных конструкций (например, именных групп), можно непосредственно записывать только в шаблонах LSPL и правилах Томита-парсера. Заметим также, что простейшая возможность задания произвольных символьных строк, включающих как буквенные, так и небуквенные символы, реализована не во всех системах.

В целом, язык LSPL является достаточно гибким средством записи лексических и грамматических свойств ЕЯ-выражений и сравним по мощности с языками систем RCO и Томита-парсер, причем с последним его объединяет намаловажная возможность явного задания условий грамматического согласования. В то же время по декларативности и наглядности лексико-синтаксические шаблоны LSPL не уступают шаблонам системы Alex и превосходят другие языки.

Первые приложения языка шаблонов LSPL

Описываемые приложения разрабатывались на базе программного комплекса поддержки языка LSPL [6], ядром которого является библиотека компонент для автоматического поиска в заданном тексте на русском языке конструкций по описывающим их LSPL-шаблонам. Поскольку автоматически выделенные языковые конструкции обычно требуют некоторой дальнейшей обработки для решения конкретных прикладных задач, программная библиотека включала также простейший механизм преобразования найденных текстовых фрагментов, в частности, нормализацию слов выделенной конструкции (приведение их к стандартной форме, которая зависит от части речи слова) однако сам язык эти возможности не поддерживал.

Первые приложения по автоматической обработке ЕЯ-текстов, разработанные на основе программного комплекса LSPL, включали набор процедур терминологического анализа, вопросно-ответную систему с логическим выводом и модуль генерации тестов для Java-программ по Javadoc-комментариям [6].

В вопросно-ответной системе вопросы задаются в виде предложений русского языка и переводятся системой в формулы логики первого порядка, при этом применяются LSPL-шаблоны вопросов, шаблоны именований сущностей, фигурирующих в вопросах, а также шаблоны описания свойств этих сущностей.

В модуле генерации тестов используются лексико-синтаксические шаблоны именований элементов программного кода (*первый параметр*, *результат* и др.), на базе которых автоматически строятся и затем применяются шаблоны конструкций, описывающих

различные аспекты поведения программного кода (например, взаимосвязь параметров и результата).

Наиболее крупным ЕЯ-приложением, созданным на базе LSPL, был комплекс процедур для автоматического терминологического анализа русскоязычных научно-технических текстов [9]. В рамках этого приложения в виде LSPL-шаблонов были формально описаны:

- Морфосинтаксические образцы терминологических словосочетаний, употребляемых в научно-технических текстах (именные группы разной структуры). Приведем пример такого шаблона:

  `A1 N1 N2<c=gen> <A1=N1> (N1)`

  описывающего, в частности, термин *гомоморфная функция ядра*.

- Словарные термины и их синонимы (из словарей по физике и информатике). Например, LSPL-шаблон

  `N1<адрес>{N2<команды,c=gen>|`
  `N2<возврата,c=gen>|N2<устройства,c=gen>}<1,1>`

  описывает три термина, начинающихся со слова *адрес*: *адрес команды*, *адрес возврата*, *адрес устройства*.

- Регулярно используемые в научно-технических текстах конструкции-определения новых терминов, а также конструкции введения терминологических синонимов. В частности, шаблон

  `NP<c=acc> 'будем' 'называть' Term1<c=ins>`

  описывает определения вида *Такие операции будем называть понятийными операциями* (внутренний шаблон `Term1` соответствует самому термину *понятийная операция*).

Однако для формализации на языке LSPL более сложных случаев употреблений терминов существующих средств языка шаблонов оказалось недостаточно, требовались дополнительные возможности, определяющие, в частности, способы обработки распознанной по шаблону конструкции – термина или комбинации терминов. Проведенное *ad hoc* расширение языка – добавление при необходимости одного или нескольких шаблонов, описывающих нужную обработку (от основного шаблона они отделяются знаком #), позволило формализовать также:

- Правила образования текстовых вариантов терминов, в частности, вариантов, возникающих в результате замены слова на однокоренное другой части речи, как в варианте вида *коллекция текстов – текстовая коллекция*; соответствующий шаблон:

  `N1 N2<c=gen> # A1 N1 <A1.st=N2.st>`

  (характеристика `st` указывает основу слова).

- Правила образования в тексте соединений (комбинаций) нескольких терминологических словосочетаний. К примеру, шаблон

```
'как' A1 ',' 'так' 'и' A2 N <A1=A2=N>
                    # A1 N <A1=N>, A2 N1 <A2=N>
```

описывает случаи соединения двух терминов вида *A N* (прилагательное-существительное) с помощью двойного союза *как, так и* (в частности, соединение *как именные, так и глагольные словосочетания* образовано из терминов *именное словосочетание* и *глагольное словосочетание*).

Для всех рассмотренных групп LSPL-шаблонов были построены процедуры распознавания и выделения в научно-технических тестах терминов разного вида и экспериментально исследованы показатели их точности и полноты [9].

Опыт применения языка LSPL в трех рассмотренных ЕЯ-приложениях показал его достаточную выразительность, а также эффективность поддерживающих его программных средств. В то же время выявилась необходимость новых средств, обеспечивающих задание в явном виде необходимого преобразования распознанной по шаблону ЕЯ-конструкции, включая генерацию новых шаблонов.

Новые средства языка LSPL

Для преобразования конструкции, выявленной по шаблону, в записи этого шаблона допускается дополнительный компонент, описывающий либо способ извлечения из конструкции ее фрагментов, либо образец построения нового шаблона. Расширенный таким образом LSPL-шаблон схематично выглядит следующим образом:

шаблон распознавания `=text>` *шаблон извлечения текста*

или *шаблон распознавания* `=pattern>` *синтезируемый шаблон* .

Тем самым LSPL-шаблон является по сути ***правилом преобразования*** ЕЯ-выражения, найденного по *шаблону распознавания*, в извлекаемый текст или же в новый шаблон (для разделения левой и правой частей правила служит слово `=text>` или `=pattern>`).

В первом случае шаблон в правой части правила описывает составные части извлекаемого текста: ими в общем случае могут быть произвольные элементы-строки, а также элементы-слова и внутренние шаблоны, использованные в шаблоне распознавания (с заданием при необходимости их новых морфологических характеристик). Возможно также указание *операции нормализации* (обозначаемой знаком *#*) элементов-слов (приведение их к форме со стандартными значениями морфологических характеристик), а также задание *операции грамматического согласования* (`~>`) извлекаемых элементов, подобной условию согласования в шаблоне распознавания.

Приведем пример правила для распознавания в тексте простой сочинительной конструкции (в любой грамматической форме, например, *высокого и длинного дома*), и извлечения из нее фрагмента

без второго прилагательного, причем в нормализованной форме и с согласованием оставшегося прилагательного (*высокий дом*):

```
A1 'и' A2 N <A1=A2=N>  =text> #A1 #N <A1~>N>
```

В случае же синтезируемого шаблона в правой части правила описывается способ построения этого шаблона из элементов распознанной конструкции. При этом кроме обычных средств языка могут быть использованы *ссылки* (помечаемые знаком $) на элементы-слова и вспомогательные шаблоны, использованные в шаблоне распознавания, а также ссылки на их морфологические признаки. К примеру, следующее правило служит для распознавания двухсловных сочетаний, начинающихся со слова *понятие* (*понятие решетки* и т.п.) и построения шаблона для поиска второго слова этого сочетания:

```
N1<понятие> N2 <c=gen> =pattern>  N <$N2.st>
```

Средства синтеза шаблонов необходимы, в частности, в процедурах терминологического анализа для генерации по распознанным в тексте комбинациям терминов шаблонов входящих в комбинацию терминов с целью дальнейшего поиска в тексте этих терминов.

Приложения на основе расширенных шаблонов LSPL

С использованием расширенного языка лексико-синтаксических шаблонов была построена система для извлечения информации из русскоязычных текстов финансовых обзоров, выпускаемых аналитическими департаментами инвестиционных компаний и публикуемых в сети Интернет [9]. Каждый из обрабатываемых текстов содержит упоминание о выпуске некоторой компанией финансовой отчетности за определенный временной период, что представляет собой извлекаемое из текста событие.

При построении системы была собрана коллекция текстов интернет-обзоров, в результате анализа которой разработан набор LSPL-шаблонов. Приведем в качестве примера фрагмент одного из текстов коллекции: *Вчера Автоваз подвел финансовые итоги за 3-й квартал 2010 года. Выручка компании выросла на 57 %, а себестоимость – на 40 %, в результате чего маржа по валовой прибыли составила приличные 12.2 %...*

Из текстов извлекались следующие атрибуты события :

– название компании, опубликовавшей отчетность (*Автоваз*);
– период, за который представлены финансовые результаты (в приведенном фрагменте – *3-й квартал 2010 года*);
– изменение выручки компании (во фрагменте – *выросла на 57%*);
– качество отчетности относительно ожиданий финансового аналитика.

Все атрибуты могли быть представлены в текстах в рамках разных контекстов. В частности, наиболее распространенными конструкциями, описывающими динамику выручки, были фразы вида:

выручка НОВАТЭКа сократилась на 58.4 %;

выручка компании увеличилась в 2010 г. по сравнению с 2009 г. на 6 %;
выручка превзошла аналогичный показатель прошлого года (+38.4 %).

Составление набора LSPL-шаблонов происходило итеративно, путем его последовательного тестирования его на коллекции текстов, уточнения и расширения набора, и в результате для описания имен компаний было составлено 8 шаблонов, для отчетного периода – 27 шаблонов, для изменения выручки – 36, а для качества отчетности – 32.

Приведем пример одного из шаблонов для распознавания фразы, выражающей факт публикации отчетности:

```
COMPANY = NAME {Av1}<0,2>{V1<опубликовать,t=past>|
  V1<обнародовать,t=past>|V1<подвести,t=past>}<1,1>
```

(вспомогательный шаблон *NAME* описывает название компании).

Извлечение системой информации из текстов финансовых обзоров происходит в несколько этапов:

1. выделение во входных текстах языковых конструкций по составленным шаблонам, с помощью программных средств LSPL;
2. извлечение из выделенных конструкций требуемой информации об атрибутах рассматриваемого события;
3. формирование из извлеченных атрибутов записи и занесение ее в базу данных.

Эксперименты, проведенные на основе построенной системы, показали довольно высокую эффективность извлечения информации: точность извлечения каждого атрибута события оказалась более 90%, а полнота в среднем превышает 75%.

Следующим приложением, реализованным на базе расширенного языка шаблонов LSPL, была система автоматизированного построения глоссариев. Глоссарий документа в норме должен содержать все основные определяемые в нем термины, включая терминологические синонимы, которые часто вводятся вместе с основным термином (например: ...*будем называть определителем, или детерминантом матрицы*...). По форме глоссарий представляет собой упорядоченный по алфавиту список глоссов – фраз вида: *Термин — Толкование*.

Для выявления определений терминов в тексте и последующего преобразования их в глоссы был взят набор шаблонов для поиска конструкций-определений терминов, разработанный ранее в рамках комплекса процедур терминологического анализа научно-технических текстов. Для того чтобы представить выявленное определение в форме глосса, потребовалось добавить к каждому из LSPL-шаблонов набора правую часть (шаблон) вида

```
=text> Term <c=nom> '-' Defin <c=nom>.
```

где *Term* – шаблон для описания структуры самого термина, а *Defin* – шаблон конструкции, определяющей термин (его толкование).

Приведём пример готового шаблона для построения глоссария:

```
'под' Term1<c=ins> ['обычно'|'здесь'|
'при' 'такой' 'формализации'] V1<пониматься,p=3,
  t=pres,m=ind>  Defin1<c=nom>  <Defin1.n=V1.n>
      =text>  Term1<c=nom> '-'Defin1<c=nom>
```

Если этот шаблон применить к тексту ...*под экономическими ресурсами понимаются все природные, людские и произведенные человеком ресурсы, которые используются для производства товаров и услуг*..., то в результате применения получим текст: э*кономические ресурсы – все природные, людские и произведенные человеком ресурсы, которые используются для производства товаров и услуг.*

Еще одним приложением, разработанным с помощью расширенного языка LSPL, была вопросно-ответная система по теории элементарных чисел. Система допускает вопросы о кратности, простоте и четности чисел, об их делителях, о наибольшем общем делителе и наименьшем общем кратном, о принадлежности числа ряду Фибоначчи и др. Вопросы задаются на русском языке, причем в рамках каждого типа вопроса допускаются различные формулировки. Например, вопрос о делителях числа может звучать совершенно по-разному: *Назови делители числа 6*, *Чему кратно число 6?*, *На что делится число 6?* и т.п.

При разработке системы было выделено 22 типа вопросов, и для каждого типа был составлен свой набор LSPL-шаблонов (в итоге получилось 73 шаблона). Приведем пример шаблона для запросов о наибольшем общем делителе:

```
['найди'|'выведи'|'какой'|'каков'|'есть''ли']
'нод'['у']['чисел'] W1 'и' W2  =text> 'нод' W1 W2
```

При распознавании вопроса по шаблону система определяет и фиксирует (в виде ключевого слова) его тип и аргументы, а затем вызывает нужную функцию, вычисляющую ответ и формирующую фразу ответа.

Заключение

В работе рассмотрен опыт использования языка лексико-синтаксических шаблонов LSPL, предназначенного для формального описания языковых конструкций в системах автоматической обработки текстов на русском языке. Основные средства языка охарактеризованы в сравнении с другими аналогичными языками.

Успешное применение языка для разработки различных ЕЯ-приложений позволило проверить его выразительные возможности и на этой основе расширить язык новыми средствами, упрощающими построение приложений. Отметим, что реализованные на базе LSPL приложения относятся к разным классам ЕЯ-систем, что позволило наметить дальнейшие пути развития его выразительной мощности.

К очевидным улучшениям относится возможность использования синонимов уже введенных в язык средств:

русскоязычных названий морфологических характеристик, общеизвестных операторов регулярных выражений `*`, `+`, `?` и др.

Более существенно введение в язык логических операций отрицания (`!`), дизъюнкции и конъюнкции, применяемых к условиям конкретизации морфологических характеристики и условиям согласования, что позволит записать, к примеру, шаблон `V<!(t=past)>`, соответствующий глаголу в любом времени, кроме прошедшего.

Насущным направлением является также совершенствование языковых средств для работы с подключаемыми словарями. В настоящий момент язык допускает словарные условия, записанные в форме обращений к булевским функциям, но не регламентирует структуру и синтаксис аргументов этих функций.

Другой точкой расширения языка может быть снятие ограничений, накладываемым LSPL-шаблоном на порядок следования элементов описываемой ЕЯ-конструкции – сейчас он жестко фиксируется. В языках с относительно свободным порядком слов, каковым является русский, это не дает возможность компактно описать такие конструкции, как глагол и его дополнение, которое может располагаться как до, так и после глагола. Введение в язык специальной операции-связки `~`, обозначающей произвольный порядок вхождения связываемых ею операндов-элементов в описываемую ЕЯ-конструкцию, позволило бы сделать язык более гибким. С использованием этого оператора запись `N ~ V` обозначает существительное и глагол, идущие в любом порядке друг за другом.

Еще одно важное направление развития языка связано с многоуровневостью ЕЯ-текста как лингвистического объекта. К основным его языковым уровням относятся уровни графематики, морфологии, синтаксиса и семантики. Средства языка LSPL в основном относятся к уровню морфологии, графематика и синтаксис представлены фрагментарно, и практически не представлен семантический уровень. Поэтому очевидна полезность новых выразительных средств, относящихся в первую очередь к графематическому уровню (например, шаблоны токенов разного вида).

Литература

1. Cunningham H., et al. JAPE: a Java Annotation Patterns Engine. (Second Edition). Technical report CS--00--10, University of Sheffield, Department of Computer Science, 2000.
2. General Architecture for Text Engineering – http://www.gate.ac.uk/
3. Grishman R. Information extraction. In: The Oxford Handbook of Computational Linguistics. Mitkov R. (ed.). Oxford University Press, 2003. p. 545-59.

4. Lexico-Syntactic Pattern Language: описание языка LSPL – http://www.lspl.ru/
5. Большакова Е.И., Баева Н.В., Бордаченкова Е.А., Васильева Н.Э., Морозов С.С. Лексико-синтаксические шаблоны в задачах автоматической обработки текстов // Компьютерная лингвистика и интеллектуальные технологии: Труды Межд. конференции Диалог ‘2007. – М.: Издательский центр РГГУ, 2007, с.70-75.
6. Большакова Е.И., Носков А.А. Система для поиска и выделения конструкций в текстах на естественном языке // Двенадцатая национальная конф. по искусств. интеллекту с международным участием (КИИ-2010): Труды конференции. Т. 4., 2010, с.63-71.
7. Большакова Е.И., Жеребцова Ю.А. Эксперименты по извлечению информации из аналитических текстов финансовых обзоров // Информационные системы для научных исследований: Сборник научных статей. Труды XV Всерос. объединенной конф. "Интернет и современное общество". Санкт-Петербург, 2012, с.190-194.
8. Ермаков А.Е. и др. RCO Pattern Extractor: компонент выделения особых объектов в тексте // Информатизация и информационная безопасность правоохранительных органов: XI Межд. научная конференция. Сборник трудов – Москва, 2003. с. 312-317.
9. Ефремова Н.Э., Большакова Е.И., Носков А.А., Антонов В.Ю. Терминологический анализ текста на основе лексико-синтаксических шаблонов // Комп. лингвистика и интеллектуальные технологии: По материалам Междунар. конференции «Диалог». Вып. 9 (16). – М.: Изд-во РГГУ, 2010, с. 124-129.
10. Жигалов В.А. и др. Система Alex как средство для многоцелевой автоматизированной обработки текстов // Труды Международного семинара Диалог’2002: "Компьютерная лингвистика и интеллектуальные технологии". М.: Наука, 2002. Т.2, С.192-208.
11. Рабчевский Е., Булатова Г., Шарафутдинов И. Формализм записи лексико-синтаксических шаблонов в задаче автоматизации процесса построения онтологий // Труды десятой Всероссийской научной конференции RCDL’2008. Дубна: ОИЯИ,2008, с.415.
12. Скатов Д.С. и др. Язык описания правил в системе лексического анализа ЕЯ-текстов DICTASCOPE TOKENIZER // Компьютерная лингвистика и интеллектуальные технологии: По матер. Межд. конф. «Диалог». Вып. 9 (16). – М.: Изд-во РГГУ, 2010, с. 442-499.
13. Томита-парсер – http://api.yandex.ru/tomita/
14. Хорошевский В.Ф. OntosMiner: семейство систем извлечения информации из мультиязычных коллекций документов // Девятая Национальная конференция по искусственному интеллекту КИИ-2004. Т. 2. – М.: Физматлит, 2004, с. 573-581.