ГЛАВА 2. МОРФОЛОГИЧЕСКИЙ И ПРЕДСИНТАКСИЧЕСКИЙ АНАЛИЗ

Настоящая глава посвящена проектированию морфологического анализа и некоторым методам снятия грамматической омонимии и построения именных групп, которые являются неотъемлемой частью современного синтаксического процессора. На нынешнем этапе развития промышленные варианты лингвистических анализаторов заканчиваются на уровне выделения именных np-групп, и по-прежнему центральным звеном таких систем остается морфология. В разделах III и IV главы рассматривается один из наиболее успешных проектов промышленного лингвистического процессора.

I. Прикладной морфологический анализ без словаря.

В 60-70 гг. все экспериментальные исследования в области машинной морфологии начинались с создания машинного словаря. Не было единого общепринятого формата и структуры такого словаря. Эти обстоятельства имели два последствия: во-первых, все алгоритмы автоматически становились словарнозависимыми, во-вторых, каждый алгоритм разрабатывался под определенный формат словаря. На современном этапе развития информационных технологий морфологический компонент стал неотъемлемой частью интеллектуальных информационно-поисковых систем (ИПС).

Основная проблема в разработке машинно-ориентированного алгоритма для лингвистических процессоров состоит в объеме исходных данных, используемых программой, то есть в объеме словарей, которые приходится составлять вручную. Исследования в этой области направлены на минимизацию исходных данных. Работы, посвященные морфологии, можно условно разделить на две категории:

1. теоретические, в некоторых представлены описания морфологических законов и формальные модели русской морфологии;
2. прикладные, описание програмно-реализованных систем с морфологическим модулем.

В теоретических работах строятся многоуровневые формальные модели морфологии, в большинстве своем, предназначенные для синтеза. Такие модели морфологического синтеза подразумевают наличие больших словарей со сложной структурой. Они описывают широкий круг морфологических явлений. Многие компоненты этих моделей избыточны для задач машинного анализа (фонетическая реализация слова, акцентная парадигма, большое число словообразовательных аффиксов).

В теоретической работе “Формальная модель русской морфологии” [Н.А.Еськова, И.Г.Бидер и др.] дается полное описание морфологических явлений русского языка и

нестандартные решения для их формализации. Перечислим важные особенности данной модели: (а) различение морфологического и (б) синтаксического рода[1]; (в) отнесение темы глагола (‘-ов-’, ‘-у-’, ‘-а-’ и т.д.) к флексии; (г) метод описания чередований для существительных и различение для супплетивных основ[2]; (д) выделения специальных признаков глагола, различные комбинации значений которых покрывают все возможные в русском языке способы видообразования (всего 32 комбинации); (е) отсечение отрицания (частицы ‘не’) у существительных и прилагательных. Недостатками такой модели является ее сложность: (а) несколько уровней представления морфологической информации, специальные грамматики для перехода с одного уровень на другой; (б) избыточность грамматических признаков, часть из которых выделены в модели для описания частных случаев.

Модели, которые используют словарь, способны дать более полный анализ словоформы (т.е. оперировать большим числом грамматических признаков). Степень точности такого анализа выше, по сравнению с моделями, которые не используют словаря. В разделе II текущей главы будет рассмотрен ряд систем морфологического анализа с использованием грамматических словарей. На пространстве реальных текстов системы, использующие словарь, часто дают сбои. Это обусловлено тем, что не существует полных словарей. Лексика языка непрерывно пополняется - появляются новые слова. Для каждой предметной области существует своя терминология, свое подмножество лексики языка, и включить в общий словарь всю существующую терминологию - невозможно. Равно как невозможно и перечислить все существующие имена и фамилии, которые имеют регулярное склонение.

Алгоритмы программ, работающих без словаря, используют вероятностно-статистические методы и лексиконы суффиксов или квази-суффиксов, основ или квази-основ, построенных эмпирически. В статье “Эмпирическое моделирование в вычислительной морфологии” [С.О.Шереметьева, С.Ниренбург, 1996] описана работающая модель морфологического анализа, не требующая объемных словарей основ открытых классов слов. Модель разработана в русле инженерной лингвистики. Модель использует следующие лексиконы:

1. Лексикон окончаний и рефлексивов;

---

[1] ***Пример:***

‘Мужчина’ : морфологический род = женский; синтаксический род = мужской.

‘Подмастерье’ : морфологический род = средний; синтаксический род = мужской.

[2] ***Пример:*** *небо’-‘небес’ (тема ‘ес’); ‘мать’- ‘матери’ (тема ‘ер’).*

2. Лексикон суффиксов;

3. Лексикон квази-корней;

4. Лексикон префиксов;

5. Лексикон баз;

6. Лексикон основ.

Каждой единице такого лексикона приписаны все возможные грамматические характеристики словоформ, частью которой может являться данная единица. Пример единицы лексикона квази-корней:

-ени-

существительное, 11, -е,

существительное, 8, -й,

глагол, -ть;

где 11, 8 - тип склонения.

Анализ словоформы в модели построен на правилах поиска и сочетания единиц разных лексиконов, что приводит к унификации гипотез.

Такой анализ не использует возможности текстов, поступающих на вход системы. По сути, предлагаемый метод сводится к эмпирическому сжатию исходного словаря словоформ. Для этого выделяются общие цепочки букв в множестве словоформ, и каждой цепочке букв приписываются все возможные значения грамматических категорий этих словоформ. Эмпирическое сжатие грамматического словаря русского языка приводит к созданию большого числа разрозненных лексиконов разной структуры, каждый из которых требует отдельной процедуры считывания данных. В статье не описана технология формирования лексиконов. Данный подход к морфологическому анализу нельзя назвать, в полной мере, бессловарным.

Похожий метод используется в работах Г.Г.Белоногова [Г.Г.Белоногов, 1984], где дается описание вероятностно-статистических методов для создания вспомогательных лексиконов на основе исходного корпуса текстов.

Все алгоритмы такого рода имеют одни и те же недостатки:

(1) не используются точные лингвистические методы анализа;

(2) большой объем лексиконов;

(3) вероятностно-статистические методы плохо работают с малой выборкой.

Точность такого анализа намного ниже, чем для систем, работающих со словарем. Эти алгоритмы не позволяют выбирать уникальные грамматические характеристики, хотя в большинстве случаев позволяют построить общую основу или квази-основу для множества словоформ и лемматизировать словоформу.

Наиболее свободная форма анализа была разработана в Чикагском Университете [J. Goldsmith, 1999]. Модель позволяет путем статистической обработки большого массива текстов, анализируя частоту встречаемости последовательности символов в словоформах, выделять множество аффиксов и корневых морфем, релевантных для заданного языка. Программа работает с большинством европейских языков, включая русский. Работа проводилась в рамках научного исследования и не получила прикладного внедрения.

В этом разделе предлагается описание модели прикладного морфологического анализа без словаря, разработанной автором диссертации в НТЦ "Система" в период с 1997 по 1998 гг. Алгоритмы морфологии построены на самообучении программы на открытых массивах реальных текстов и совмещают два подхода: лингвистический - формализованная грамматика для построения морфологических гипотез и математический - метод корреляции, позволяющий унифицировать морфологическую гипотезу. Морфологический анализ без словаря является центральной компонентой системы автоматической индексации текстовой базы данных (БД), реализованной в СУБД Oracle8i. Выходным результатом системы является автоматически построенный грамматический словарь основ и связанный с ним индекс документов, предназначенный для полнотекстового поиска по БД.

Сущность интеллекта состоит в способности принимать разумные решения в условиях отсутствия полноты данных и фактов [M. Boden, 1990]. Интеллектуальность системы повышается с уменьшением объема статической информации, используемой в процессе анализа данных. В нашем случае, речь идет об использовании лингвистической информации при морфологическом анализе в задачах автоматической индексации текстовых БД. Ниже будут выделены основные критерии, отличающие большинство интеллектуальных систем, которых придерживается описываемый процессор автоиндексации текстов:

- Способность системы объяснить каждой шаг принятых решений. В процессе анализа не используются вероятностные и статистические методы.
- Использование правил и свойств, характеризующих данный предмет анализа. Для построения морфологических гипотез словоформ используется формализованная грамматика и то свойство русского языка, что большая часть грамматических категорий в русском вычисляется из флексии.
- Модульность системы, которая обеспечивает эффективное изменение и пополнение правил и свойств, а также задает возможность настраивать анализатор на другие естественные языки с развитой морфологией.

- Множественность интерпретаций. Анализатор оставляет все омонимы значений словоизменительных категорий.
- Самообучаемость и механизм исправления принятых ранее неверных решений. Объем прочитанных текстов пополняет число словоформ, используемых в процессе анализа, тем самым повышая точность морфологического анализа и позволяя корректировать неправильно построенные основы и значения их грамматических категорий.
- Моделирование интеллектуального поведения человека. В данном случае, речь идет о попытке эмулировать размышления человека, изучающего иностранный язык, перед которым стоит задача классифицировать слова данного языка, в условиях, когда в его распоряжении находится большой массив текстов, некоторые знания о морфологии языка и отсутствует словарь языка, на котором написаны тексты. Надо сказать, что при разработке алгоритмов не ставилось задачи опровергнуть мысленный эксперимент Джона Сёрля “Китайская комната” [см. гл. 1, раздел II].

Модель будет рассмотрена на уровне общего описания процессора - взаимодействие его модулей и функциональная схема алгоритма морфологического анализа [И. Ножов, 2000].

Схема процесса автоматической индексации представлена на рис.1: на вход процесса автоиндексации поступает все множество текстов, хранящихся в базе данных, на выходе формируется словарь основ и таблица соответствий (текст ⇔ основа), которая отображает поток индексированных текстов.

Блоки, которые осуществляют процесс автоиндексации, представлены на рис.2.

Процессы (рис.2):

1. Графематический анализ.
2. Морфологический анализ.

На рис.3 показана схема таблиц для хранения потоков данных, сформированных процессами графематического и морфологического анализа.

Потоки данных (рис.3):

1. Тексты;
2. Полные словоформы;
3. Аббревиатуры;
4. Цифровые и символьные комплексы;
5. Основы и значения их грамматических категорий;

Основная цель графематического блока получить выборку полных словоформ из массива текстов БД. Графематический анализ работает с внешним представлением текста

и использует таблицу стоп-слов. В этой таблице хранятся цифры, спецсимволы и частотные слова языка, нерелевантные для поиска по текстам.

Графематический анализ выполняет три функции:

1. отсечение стоп-слов в тексте;
2. разбиение данных на три потока;
3. индексация каждого потока.

Единицей графематического анализа является цепочка символов, выделенная с двух сторон пробелами. Выделенная цепочка символов подвергается последовательной обработке эвристическими правилами: отсечь знаки пунктуации, проверить присутствие гласных внутри цепочки, чередование верхнего и нижнего регистров и т.д. В зависимости от результатов обработки полученная цепочка символов направляется в один из трех потоков данных:

— цифровые и символьные комплексы (‘кг’, ‘ст.’, ’12.01.99’);

— аббревиатуры - названия государств, организаций, предприятий (‘СССР’, ‘ЮНЕСКО’, ‘ДорСтройСервис’);

— полные словоформы;

Каждой записи из любого потока ставятся в соответствие коды документов, в которых она встретилась. Первых два потока данных считаются проиндексированными, причем только аббревиатуры являются релевантным поисковым образом. Графематику можно считать лишь вспомогательным звеном для морфологического анализа. Графематический и морфологический процессы способны проиндексировать массивы текстов независимо от предметной области конкретной базы данных.

Полные словоформы поступают на вход морфологического анализа, цель которого разбить все множество словоформ на подмножества по признаку принадлежности к той или иной лексеме[3], привести все элементы каждого такого подмножества к уникальной основе, однозначно определить грамматические характеристики лексемы и проиндексировать тексты по встретившимся в них основам.

[3] *Лексема - это множество словоформ, отличающихся друг от друга только словоизменительными значениями [И. Мельчук, 1997].*

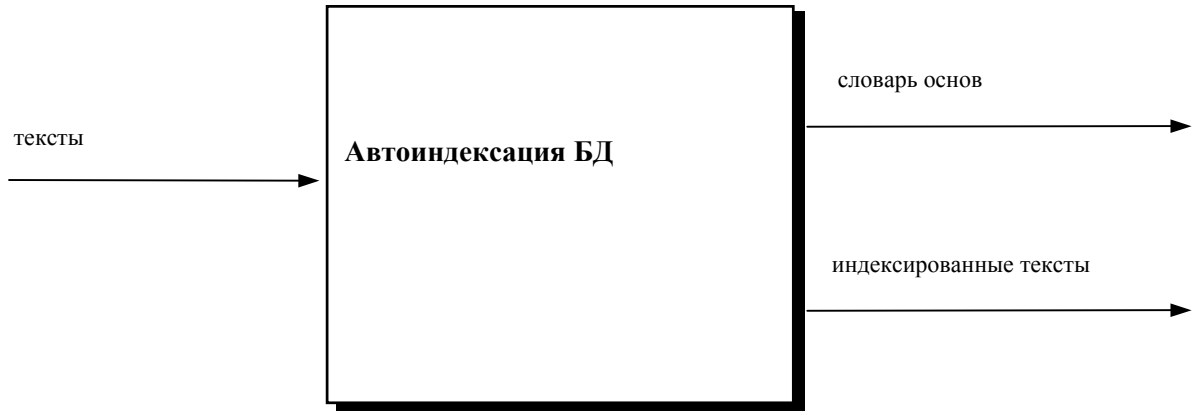
тексты
Автоиндексация БД
словарь основ
индексированные тексты

рис.1

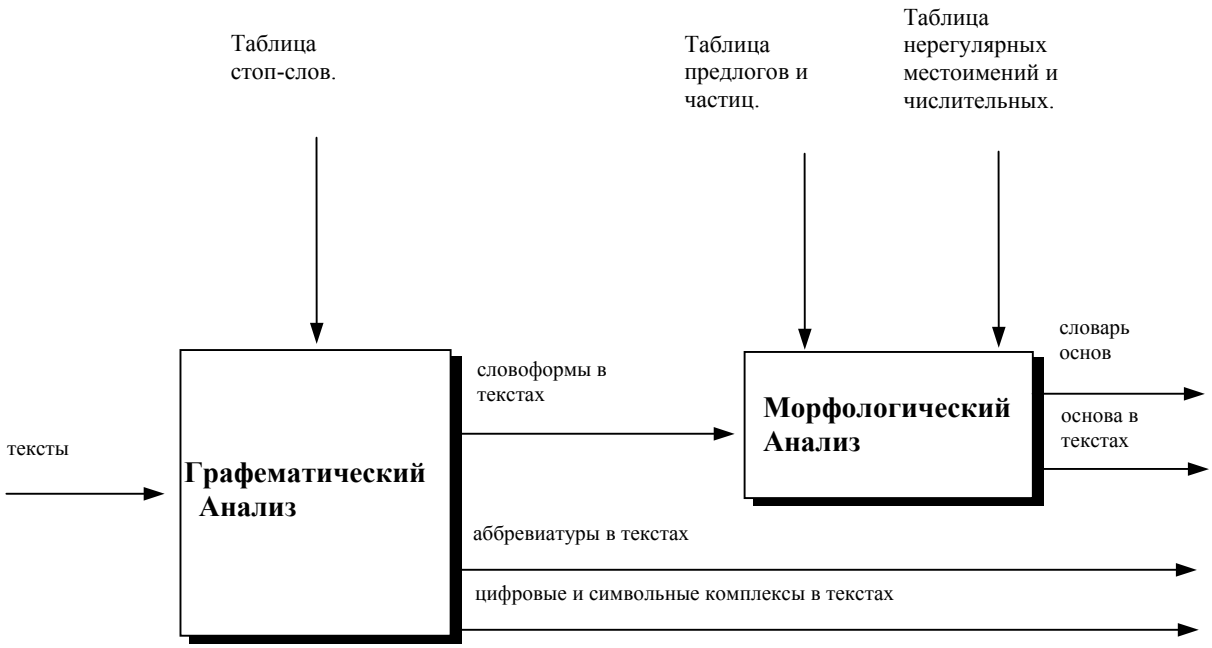
Таблица
стоп-слов.
Таблица
предлогов и
частиц.
Таблица
нерегулярных
местоимений и
числительных.
тексты
Графематический
Анализ
словоформы в
текстах
Морфологический
Анализ
словарь
основ
основа в
текстах
аббревиатуры в текстах
цифровые и символьные комплексы в текстах

рис.2

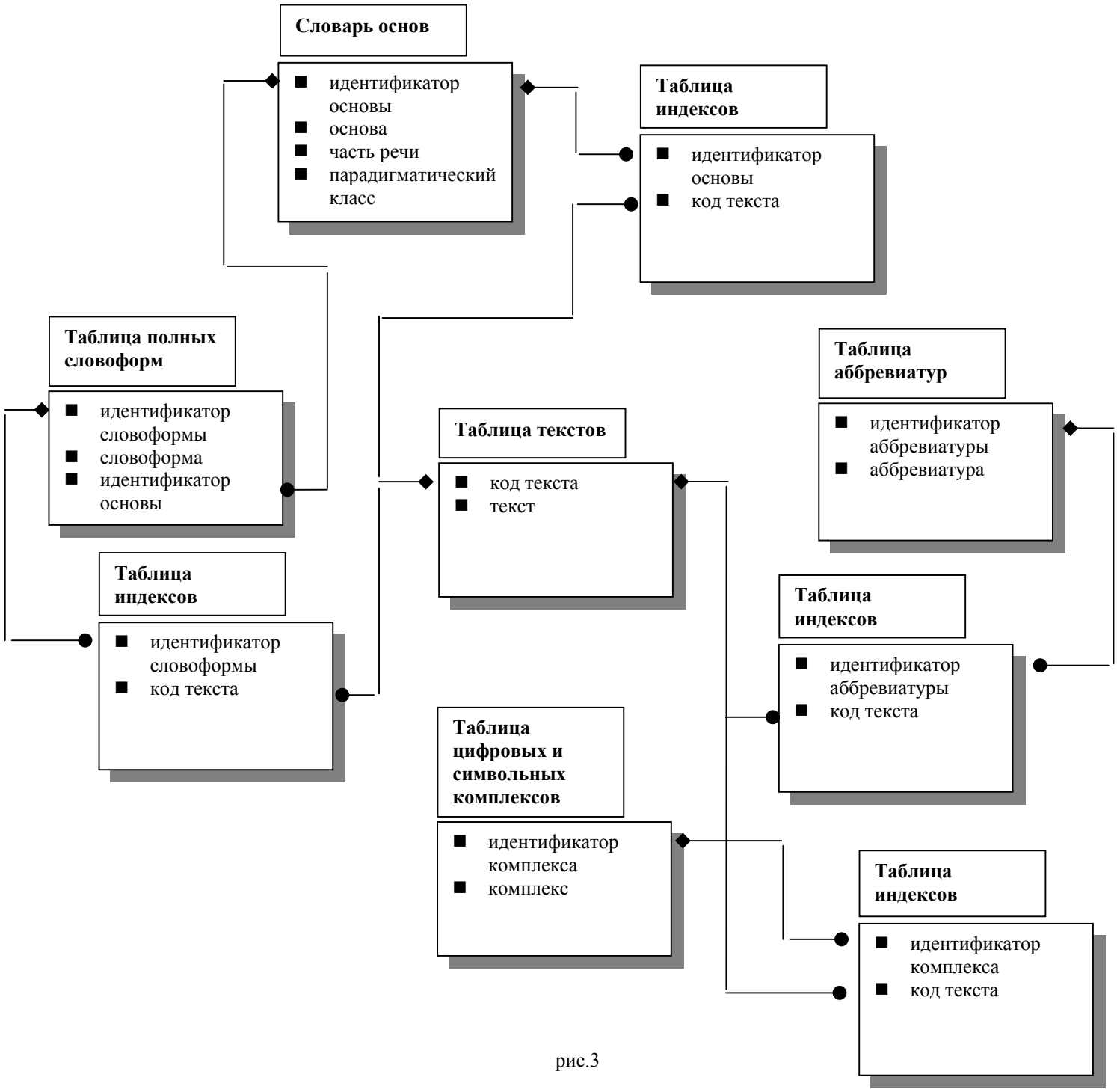
Словарь основ
идентификатор основы
основа
часть речи
парадигматический класс
Таблица индексов
идентификатор основы
код текста
Таблица полных словоформ
идентификатор словоформы
словоформа
идентификатор основы
Таблица текстов
код текста
текст
Таблица аббревиатур
идентификатор аббревиатуры
аббревиатура
Таблица индексов
идентификатор словоформы
код текста
Таблица индексов
идентификатор аббревиатуры
код текста
Таблица цифровых и символьных комплексов
идентификатор комплекса
комплекс
Таблица индексов
идентификатор комплекса
код текста

рис.3

Блок морфологического анализа использует минимальный объем исходной информации:

— таблицу предлогов;

— таблицу местоимений и числительных, имеющих нерегулярное склонение.

На выходе морфологического анализа формируется словарь основ данной БД, уникальность записи в таком словаре задается тройкой значений [основа, часть речи, парадигматический класс]. Морфологический анализ состоит из трех модулей и соблюдает определенную последовательность действий.

Первый модуль содержит статический массив флексий и правила формализованной грамматики русской морфологии, построенной на основе работ А.Зализняка [А.Зализняк, 1980]. Выделение парадигматических классов в модели полностью соответствует парадигматическим классам в словаре А.Зализняка. Это - восемь типов склонения существительных и прилагательных и шестнадцать типов парадигмы глагола, которым соответствует первое или второе спряжение. В словаре А.Зализняка глагольная тема (*'ов', 'у'* и т.д.) входит в окончание глагола. В нашем случае вводится термин расширенная флексия глагола. Расширенной флексией глагола называется конкатенация чередующейся глагольной темы и флексии.

Данный модуль может быть заменен формализованной морфологией любого другого флективного языка. Методы, описанные в модулях два и три, являются универсальными, независящими от языка.

Второй модуль, используя правила формализованной грамматики, позволяет строить морфологическое дерево словоформы, в узлах которого хранятся все возможные гипотезы об основах и значениях грамматических категорий словоформы. Морфологические правила делятся на два класса. Первый класс правил, которые порождают некоторые грамматические характеристики для гипотез, и второй класс правил накладывает определенные ограничения на гипотезы. Пример правил первого класса: *если гипотеза об основе оканчивается на согласную ряда {'к', 'г', 'х'}, то тип склонения равен трем* или *если исходная словоформа не оканчивается на гласную, то построить гипотезу о существительном с нуль-флексией.* Пример правил второго класса: *если гипотеза о флексии равна 'ет' [3 лицо, ед. ч.] или 'ю' [1 лицо, ед. ч.], и гипотеза об основе оканчивается на сегмент первой ступени чередования* [А.Зализняк, 1980], *то гипотеза о глаголе не верна.*

Традиционно в синтаксических и семантических теориях используется представление языковой структуры с помощью деревьев. В описываемой системе, пожалуй, впервые данный формализм оправдано был применен к морфологии.

Третий модуль содержит метод подбора словоформ на одну лексему[4], то есть выбор коррелятов для дерева исходной словоформы. После того, как набраны корреляты, для каждой словоформы также строится морфологическое дерево всех возможных гипотез, в результате чего образуется "лес деревьев" [Ф.Харари, 1973]. Метод корреляции[5] осуществляет сравнение морфологических деревьев внутри леса и унификацию гипотез. Корреляция проводится по гипотезам основ и значениям классифицирующих грамматических категорий, таких как часть речи, парадигматический класс, спряжение глаголов и род существительных. Значения словоизменительных категорий в корреляции не участвуют. Во время работы корреляции происходит удаление ложных гипотез: ветвей дерева или полного дерева коррелята. Этот модуль позволяет построить уникальную гипотезу об основе и значениях ее грамматических категорий для всех словоформ одной лексемы, найденных в текстах. Метод корреляции очищает лес от ложных коррелятов, оставляя, таким образом, только словоформы, принадлежащие одной лексеме. Уникальная основа, единая для всех словоформ, участвовавших в корреляции, значение части речи и парадигматического класса добавляются в словарь основ. По сути, основа в словаре репрезентирует лексему.

Для унификации гипотезы метод корреляции использует матрицы корреляций. Лесом называется множество деревьев словоформ $F = \{T_1,..,T_j,..,Tn\}$. Множество всех построенных гипотез об основе в F обозначим $U = \{s_1,..,s_i,..,s_m\}$. Параметром корреляции t называется значение грамматической категории. Матрицей корреляции $A(t) = \|a_{ij}\|$ леса F с m гипотезами об основах и n деревьями словоформ называется $(m \times n)$-матрица, в которой $a_{ij} = 1$, если заданный параметр корреляции t определен для s_i в T_j, и $a_{ij} = 0$ в противном случае.

В процессе корреляции отдается приоритет гипотезам исходной словоформы, на основе которых подбираются корреляты, что позволяет избежать ситуации, когда лес вырождается в пустое множество. Число матриц корреляции внутри одного типа корреляции определяется по числу возможных значений грамматической категории: так, в процессе корреляции по роду существительных для русского будет построено три матрицы, соответствующие трем возможно задействованным в деревьях значениям грамматического рода. Для каждой матрицы корреляции находится

---

[4] *Словоформы, которые гипотетически принадлежат одной лексеме, для сокращения записи мы будем называть "словоформы на одну лексему" [прим. автора].*

[5] Данный метод корреляции был разработан специально для задачи морфологического анализа и не имеет ничего общего с его вероятностно-статистическим аналогом, предназначенным для решения других задач [прим. автора].

$$k = \max_{i:a_{i1}\neq 0} \sum_{j=1}^{n} a_{ij}$$

после чего из множества значений k внутри одного типа корреляции также выбирается максимальное значение, которое и соответствует унифицированной гипотезе. Узлы не получившие максимального значения удаляются из деревьев словоформ. Условие $a_{i1} \neq 0$ задает приоритет гипотезам дерева исходной словоформы T_1.

Допустим в прочитанных программой текстах было подобрано два коррелята для исходной словоформы W_1, тогда лес F состоит из трех деревьев словоформ W_1, W_2 и W_3 (рис.4):

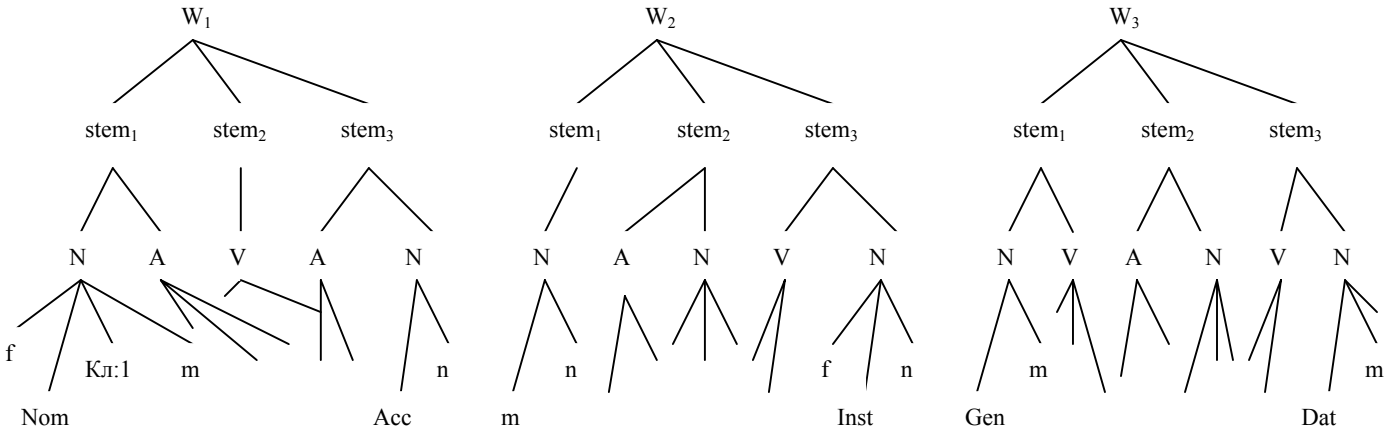

Рис.4

Корреляция по части речи:
матрица корреляции значение k максимальное значение внутри типа корреляции

$$\text{Noun} \begin{bmatrix} 111 \\ 011 \\ 111 \end{bmatrix} \bar{1} = \begin{bmatrix} 3 \\ 2 \\ 3 \end{bmatrix} \Rightarrow \begin{bmatrix} 3 & 3 \\ stem1 & stem3 \end{bmatrix}$$

$$\text{Adj} \begin{bmatrix} 100 \\ 011 \\ 100 \end{bmatrix} \bar{1} = \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} \Rightarrow \begin{bmatrix} 1 & 1 \\ stem1 & stem3 \end{bmatrix} \text{ --- } [3, 3, 1, 1, 1] \Rightarrow \text{Noun} \begin{bmatrix} 3 & 3 \\ stem1 & stem3 \end{bmatrix}$$

$$\text{V} \begin{bmatrix} 001 \\ 100 \\ 011 \end{bmatrix} \bar{1} = \begin{bmatrix} 1 \\ 1 \\ 2 \end{bmatrix} \Rightarrow \begin{bmatrix} 1 \\ stem2 \end{bmatrix}$$

Удаляются ложные узлы деревьев словоформ леса F (рис. 5):

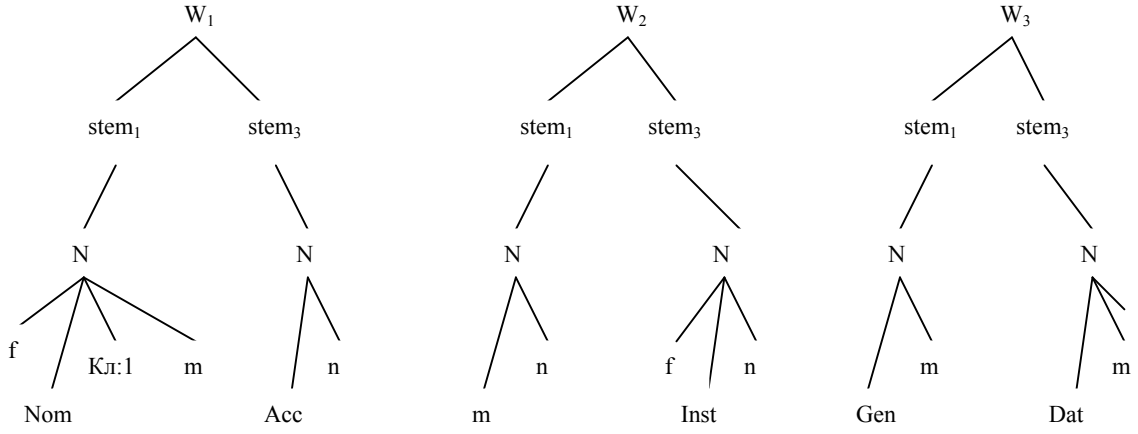

Рис.5

Корреляция по роду:
матрица корреляции значение k максимальное значение внутри типа корреляции

$$m \begin{bmatrix} 111 \\ 001 \end{bmatrix} \bar{1} = \begin{bmatrix} 3 \\ 1 \end{bmatrix} \Rightarrow \begin{bmatrix} 3 \\ stem1 \end{bmatrix}$$

$$n \begin{bmatrix} 010 \\ 110 \end{bmatrix} \bar{1} = \begin{bmatrix} 1 \\ 2 \end{bmatrix} \Rightarrow \begin{bmatrix} 2 \\ stem3 \end{bmatrix} \quad \text{--- } [3, 2, 1] \Rightarrow m \begin{bmatrix} 3 \\ stem1 \end{bmatrix}$$

$$f \begin{bmatrix} 100 \\ 010 \end{bmatrix} \bar{1} = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \Rightarrow \begin{bmatrix} 1 \\ stem1 \end{bmatrix}$$

После завершения корреляции по роду и удаления не получивших максимального значения узлов гипотеза унифицирована: W_1[$stem_1$[N[Кл:1, m, Nom, …]]]; W_2[$stem_1$[N[m, …]]]; W_3[$stem_1$[N[m, Gen, …]]].

Часто задаваемый вопрос - почему в качестве формализма выбраны деревья, а не кортежи. Деревья позволяют сделать метод корреляции универсальным, независимым от выбранного для анализа естественного языка. Как видно из примеров, ширина дерева произвольна, а высота фиксирована и равна трем для русского языка. Высота дерева, также как и ширина, может изменяться при переходе от одного анализируемого языка к другому и определяется морфологической грамматикой, т.е. существующими зависимостями между грамматическими категориями и их показателями в каждом конкретном языке, что делает использование кортежей затруднительным, а «древесный»

формализм сохраняет независимость метода корреляции от морфологических правил рассматриваемого языка.

Рассмотрим реальный пример. Визуальный интерфейс программы морфологического анализа позволяет наблюдать состояние леса до корреляции и после, как это показано на примере анализа глагола ‘текут’ (Рис.6 и Рис.7).

Подбор коррелятов и построение леса деревьев возможных гипотез для глагола ‘текут’:

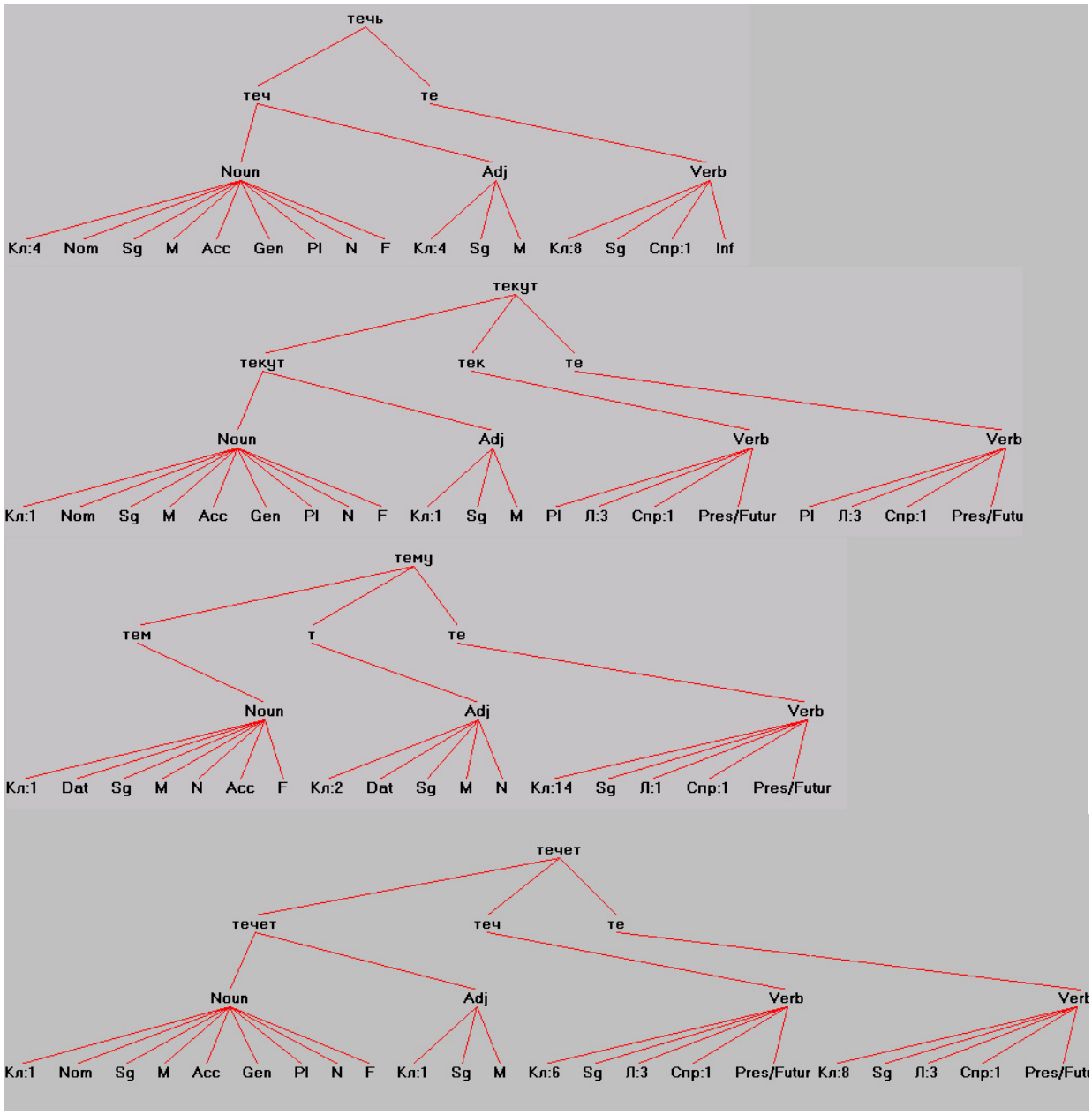

рис.6

Применение метода корреляции (унификация гипотезы и удаление ложных коррелятов):

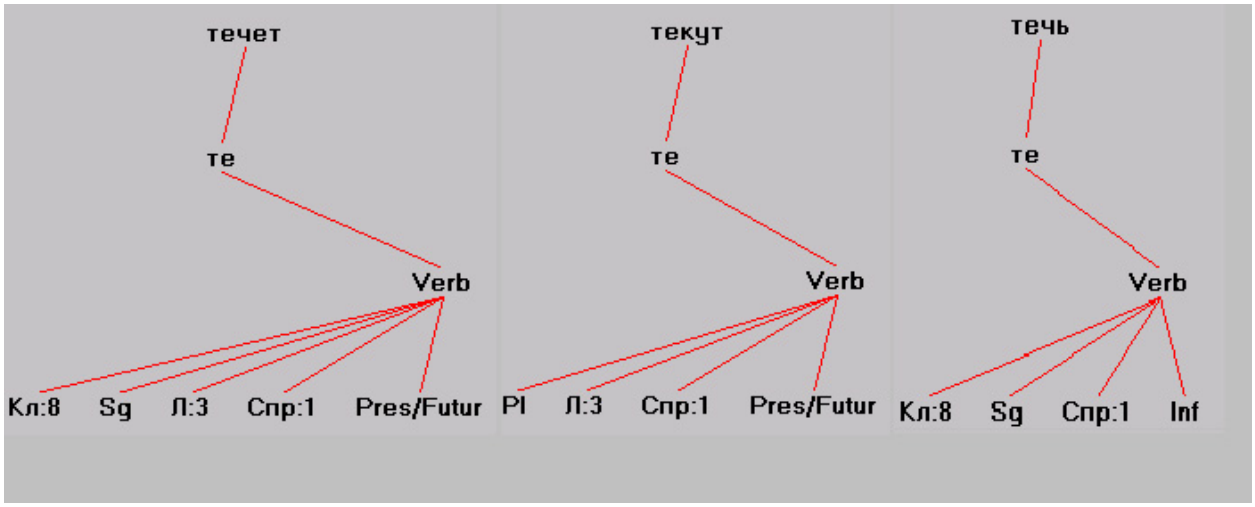

рис.7

В результате остается три дерева коррелятов с уникальными гипотезами об основе, части речи и грамматических характеристиках.

Последовательность шагов (Д1..Д13) алгоритма морфологического анализа без словаря представлена на рис.8.

Общая схема алгоритма.

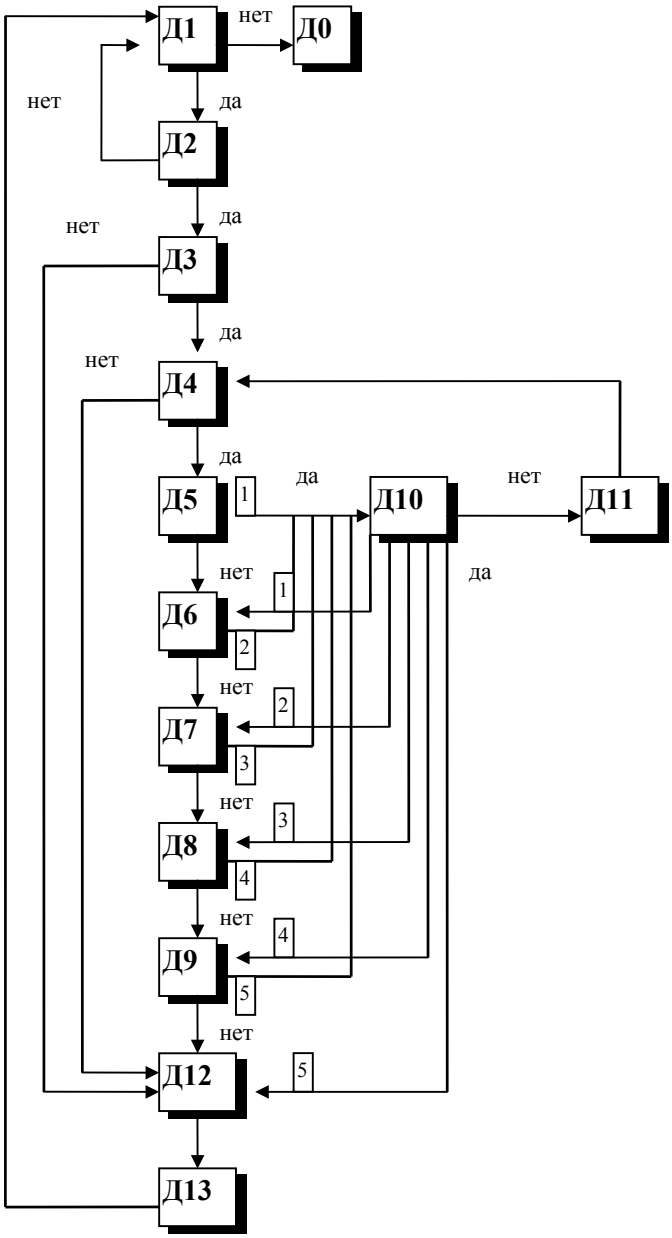

рис.4

Д0. Выход из программы.

Д1. Выбрать из таблицы полных словоформ (рис.3) непроиндексированную словоформу, то есть словоформу, для которой еще не построена основа (ДА: словоформа выбрана; НЕТ: все словоформы в таблице проиндексированы).

Д2. Проверить, что данная словоформа не является предлогом или местоимением. Построить дерево всех возможных гипотез для данной словоформы. (ДА: не является; НЕТ: является)

Д3. Выбрать из таблицы полных словоформ (рис.3) словоформы на одну лексему. Создать список коррелятов.

(ДА: корреляты выбраны; НЕТ: список коррелятов пуст)

Д4. Если список коррелятов непустой, то построить деревья всех возможных гипотез для каждого коррелята.

Д5[6]. Провести корреляцию по гипотезам основ.

Д6. Провести корреляцию по значениям части речи.

Д7. Провести корреляцию по значениям спряжения глагола.

Д8. Провести корреляцию по значениям рода существительных.

Д9. Провести корреляцию по значениям парадигматического класса.

Д10. Проверить, что корреляция не привела к удалению полного дерева (дерева коррелята) из леса. (ДА: не привела; НЕТ: привела)

Д11. Удалить ложный коррелят из списка коррелятов.

Д12. Выбрать уникальную основу и ряд грамматических характеристик к данной основе. Проиндексировать тексты, то есть выбрать для построившейся тройки [основа, часть речи, парадигматический класс] коды текстов, в которых встретились словоформы, принадлежащие данной основе.

Д13. Применить метод распределения элементов пересеченных множеств коррелятов.

Несмотря на появление объемных лексиконов для многих европейских языков и все возрастающую популярность словарного анализа, системы морфологического анализа без словаря не теряют своего прикладного значения. В задачах автоматической индексации изложенные выше алгоритмы позволяют формировать грамматические словари, являющиеся точным отображением лексики проиндексированных документов.

[6] Для Д5 - Д9 ДА: корреляция прошла успешно, то есть в деревьях словоформ были обнаружены ложные ветви и удалены; НЕТ: корреляция прошла неуспешно, то есть ложных ветвей не обнаружено. Цифровые индексы на стрелках задают маршрут продвижения по схеме, то есть индекс стрелки выхода из блока Д10 должен совпасть с индексом стрелки входа.

Бессловарная морфология сохраняет свою актуальность в задачах автоматического пополнения лексиконов. Точность такого анализа выше, чем стандартная процедура предсказания по конечной последовательности символов в слове [см. раздел II]. Использование деревьев для представление морфологической структуры словоформы и унификация гипотезы роднит задачи морфологического и синтаксического анализа, демонстрируя общность формализма и алгоритмических методов на разных уровнях лингвистического анализа. Тестирование программы, разработанной на основе полученной методики, показало работоспособность предложенной системы автоматической индексации. Метод корреляции, разработанный для настоящей задачи, позволяет выбирать уникальные гипотезы и строить словарь основ при сравнительно небольшой выборке.

II. Проектирование словарной морфологии.

Существует два базовых подхода к проектированию морфологических машинных словарей (лексиконов) для флективных языков. Первый копирует академическую лингвистическую модель описания, где выделяются основные парадигматические классы, соответствующие типу склонения и спряжения, и правила регулярных альтернаций (фонетических чередований), а нерегулярные формы (например, сильные глаголы в немецком и английском языках) задаются перечислением. Такого типа лексиконы для русского языка составляются на базе модели грамматического словаря А.Зализняка, разрабатывая 8 классов именного склонения и 16 глагольного спряжения, а чередования в основе и глагольной темы выносятся в отдельное множество пост-морфологических правил альтернаций. Второй подход рассматривает любого вида регулярное и нерегулярное чередование как часть расширенной псевдо-флексии (в таком случае, основа словоформы ‘день’ – ‘д’, а флексия – ‘-ень’; для словоформы ‘песок’: ‘пес’ и ‘-ок’). В подобной модели описания число парадигматических классов для русского языка возрастает до 3000, но рост числа классов при проектировании компенсируется однородностью лексикона и отсутствием как исключений, так и правил альтернаций.

Внутреннее устройство лексиконов первого и второго типов не влияет ни на процесс лемматизации – приведение словоформы к нормальной форме слова, репрезентирующей лексему, ни на морфанализ – определение граммем словоформы. Анализаторы, построенные на разных типах лексиконов, могут одинаково эффективно использоваться как для морфологического анализа, так и для синтеза.

Первый подход к проектированию лексиконов для построения морфологических анализаторов европейских и восточных языков был применен в научно-

исследовательском центре Xerox (Гренобль) в середине 90-ых, а позже усовершенствован и доведен до промышленного использования в исследовательских отделах Inxight Software (Санта-Клара, США и Антверпен, Бельгия) в 2000-2002 гг. Конечный продукт Inxight LinguistX Platform 3.5 включает в себя морфологии 26 языков: 5 восточных (арабский, корейский, японский, etc.) и 21 европейский (английский, голландский, испанский, русский, etc.). Наиболее разработанные языковые модули, такие как английский, немецкий и русский, имеют четыре уровня текстового анализа: tokenizer – графематика, осуществляющая деление исходного текста на предложения и словоформы; stemmer – лемматизация входных словоформ; tagger – снятие морфологической омонимии и унификация значений грамматических характеристик; и np-grouper – синтаксическое выделение именных составляющих NP из текстов.

Морфологии языков в LxPlatform состоят из двух компонент: (1) лексикон, в котором хранятся леммы (нормальная форма слова), а также парадигмы и значения их грамматических категорий; (2) множество правил альтернаций и орфографических правил. Лексикон состоит из подлексиконов (sublexicons), делящихся по селективным признакам и парадигматическим классам. Структура подлексиконов образует связанный граф, в вершине которого стоит корневой (root) лексикон, начинающий анализ входного слова [Lauri Karttunen, 1993]. Все правила второго компонента морфологии записываются на языке регулярных выражений [XRCE MLTT, 1995]. Технология анализа построена на разновидности конечных автоматов FST (finite-state transducer). FST называется автомат, в котором каждый переход между состояниями в сети (network) имеет выходную помету в дополнение к входной [XRCE MLTT, 1995]. Исходный морфологический лексикон компилируется в lexicon transducer, а компонент правил - в two-level rule transducer. Результирующий лексический конечный автомат, т.е. полное морфологическое представление языка, - lexical transducer - получается композицией lexicon transducer и rule transducer [Lauri Karttunen, 1994]. Sigma называется символьный алфавит конечного автомата [Finite-State Network, 1995]. Sigma лексического FST состоит из алфавита анализируемого естественного языка и специальных грамматических помет (tags), выражающих значение селективных признаков и граммем (+Verb – глагол, +Active – активный залог, +P1 – 1-ое лицо, +Pl – множественное число, etc.).

Построим содержащий описание глаголов ‘вписывать’ и ‘восторжествовать’ фрагмент морфологического лексикона для русского языка:

```
LEXICON Root
        Nouns; Verbs;
LEXICON Nouns
        ……
```

```
LEXICON Verbs;
        водить+Verb+Imperf:во(д/ж)ь         V2;
        вписывать+Verb+Perf:впи(с/ш)        V1;
LEXICON V1;
        +Inf+Active:^Нать  #;
        +Imperf+Inf+Passive:^Наться  #;
        +Ind+NotPast+P1+Sg+Active:^Sьу  #;
        +Ind+NotPast+P2+Sg+Active:^Sьэшь  #;
        +Ind+NotPast+P3+Sg+Active:^Sьэт  #;
        +Ind+NotPast+P1+Pl+Active:^Sьэм  #;
        +Ind+Past+Sg+Masc+Active:^Нал  #;
        +Ind+Past+Sg+Fem+Active:^Нала  #;
        +Ind+Past+Sg+Neut+Active:^Нало  #;
        ……..
LEXICON V2;
        +Inf+Active:^Нbыть  #;
        +Imperf+Inf+Passive:^Нbыться  #;
        +Ind+NotPast+P1+Sg+Active:^Sьу  #;
        +Ind+NotPast+P2+Sg+Active:^Нbышь  #;
        +Ind+NotPast+P3+Sg+Active:^Нbыт  #;
        +Ind+NotPast+P1+Pl+Active:^Нbым  #;
        +Ind+Past+Sg+Masc+Active:^Нbыл  #;
        +Ind+Past+Sg+Fem+Active:^Нbыла  #;
        +Ind+Past+Sg+Neut+Active:^Нbыло  #;
        ……
```

Корневой лексикон осуществляет вызов подлексиконов. Выражения в лексиконах представляют собой пару форм: лексическая (lexical) и поверхностная (surface) формы, разделенные двоеточием. Строящий FST компилятор интерпретирует такую пару как регулярное отношение. Решетка ‘#’ маркирует конечное состояние. Уникальность пути переходов в сети конечного автомата дает однозначность морфологической интерпретации. Приводящие в конечное состояние варианты пути в сети FST задают множественность интерпретаций для поверхностной формы, что соответствует морфологической омонимии. Так, поверхностная форма ‘стекло’ получит две лексических формы: ‘стекло+Noun…’ и ‘стекать+Verb…’. Так, лексическая форма ‘вписывать+Verb+Perf+Ind+NotPast+P3+Sg+Active’ соответствует поверхностной форме ‘впи(с/ш) ^Sьэт’.

Используя регулярные выражения [подробнее см. раздел IV], построим фрагмент компонента правил альтернации и орфографических правил для русского языка:

```
[ %(с%/ш%)->ш,
  %(д%/ж%)->ж,
|| _ ?* %^S ];

[ %(с%/ш%)->с,
  %(д%/ж%)->д,
|| _ ?* %^Н ];
```

[%^H->0, %^S->0];

[ь ь -> ь, й ь -> й];

[[ь|й]а->я, [ь|й]у->ю, [ь|й]э->е, [ь|й]ы->и];

Процент ‘%’, поставленный перед символом, переводит зарезервированный оператор в языке регулярных выражений в простой символ алфавита. ‘?’ – любой (any) символ; ‘*’ – звезда Клини; бинарный оператор ‘A -> B’ осуществляет замещение последовательности символов в левой части выражения на последовательность в правой части; выражение ‘A -> B || L _ R’ является операцией замещения, ограниченной по контексту, т.е. строке A предшествует L, и R следует непосредственно за A; ‘|’ означает дизъюнкцию. Таким образом, после применения правил к строке ‘впи(с/ш) ^Sьэт’ поверхностная форма примет свой окончательный вид: ‘впишет’.

Для английского языка в LxPlatform был разработан лексикон, включающий информацию об активном словообразовании. Модуль деривации для русской морфологии в LxPlatform отсутствует. Реализация морфологического анализатора на основе технологии конечных автоматов позволяет достичь максимальной скорости анализа.

Морфологический лексикон проекта Диалинг спроектирован с использованием второго подхода, где основа представлена неизменяемой частью слова, а все регулярные и нерегулярные чередования (в том числе и в корневой морфеме) являются частью расширенной псевдо-флексии. Лексикон насчитывает свыше 3000 парадигматических классов. Идеология морфологического анализа заимствована из работ Ж.Г.Аношкиной [Ж.Г.Аношкина, 1995].

Описание парадигмы лексемы представляет собой множество пар [псевдо-флексия, набор аношкинских кодов]. Аношкинским кодом называется уникальный двухбуквенный идентификатор, который соответствует некоторой комбинации значений селективных признаков и граммем. Конечное множество аношкинских кодов исчисляет все встречающиеся в данном языке комбинации морфологических характеристик. Всего в морфологическом анализаторе русского языка системы Диалинг насчитывается 870 таких кодов.

Приведем фрагмент описания парадигмы для лексемы ‘рукоплескать’:

1740 %СКАТЬ*ка%СКАВШАЯ*мз%ЩУ*кб%ЩУТ*кж%ЩУЩЕГО*лблглп%….

……

РУКОПЛЕ 1740

‘Рукопле’ – основа слова в лексиконе; ‘1740’ – уникальный идентификатор парадигматического класса; ‘%’ маркирует начало псевдо-флексии; ‘*’ маркирует начало

аношкинского кода; ‘ка’, ‘кб’, ‘лб’, ‘лг’, etc. – код. В таблице приведена расшифровка аношкинских кодов, использованных в примере:

код	часть речи	словообразовательные характеристики	словоизменительные характеристики	Пример
ка	Г	нс,нп	дст,инф	рукоплескать, расти
мз	Г	нс,нп,прч	прш,дст,ед,жр, им	рукоплескавшая, росшая
кб	Г	нс,нп	дст,нст,1л,ед	рукоплещу, расту
кж	Г	нс,нп	дст,нст,3л,мн	рукоплещут, растут
лб	Г	нс,нп,прч	нст,дст,ед,мр, рд	рукоплещущего, растущего
лг	Г	нс,нп,прч	нст,дст,ед,мр, вн	рукоплещущего,растущий
лп	Г	нс,нп,прч	нст,дст,ед,ср, рд	рукоплещущего,растущего

Также в множество аношкинских кодов морфологического анализатора Диалинг включены специальные коды для аналитических форм глагола, которые строятся в системе на этапе синтаксического анализа. Комбинация значений морфологических характеристик для аналитической формы глагола получается путем объединения исходных характеристик всех составляющих аналитической формы. Примеры такого кода:

Ил	П	нс,нп,	буд,мр,3л,ед,кр	будет умен
Юа	Г	нс,пе,	дпр,нст,жр,ед,кр	будучи умна

Система морфологического анализа Информэлектро, разработанная в начале 70-ых гг. в секторе (затем отделе) Д.Г.Лахути группой лингвистов под руководством Г.А.Лесскиса, является одной из первых версий машинной морфологии. Морфологический лексикон Информэлектро также можно отнести к моделям второго типа. Не используя правил альтернаций, для лексем, имеющих более одной основы, в словарь вводятся все ее основы или словоформы так, чтобы минимальным числом единиц обеспечить идентификацию всей парадигмы данной лексемы. Например, для слова «станок» вводится основа ‘станк-’ и словоформа ‘станок’. Отличительной особенностью лексикона является примитивная модель управления, которая определяется в статьях лексикона для лексем, имеющих синтаксическое управление. Примитивная модель управления (ПМУ) может принимать следующие грамматические значения: (1) управление предлогом; (2) управление родительным падежом; (3) управление дательным падежом; (4) управление винительным падежом; (5) управление творительным падежом; (6) управление предложным падежом; (7) управление подчинительным союзом; (8) управление инфинитивом. Так, для глагола ‘увидеть’ ПМУ=4,7.

В морфологических анализаторах Диалинг и Информэлектро предсказание значений селективных признаков и граммем словоформ, ненайденных в словаре, устроено

однотипно. Если входная словоформа не была найдена в словаре, то используется алгоритм предсказания, который ищет в словаре словоформу, максимально совпадающую с конца со входной словоформой [А. Сокирко, 2001] (так называемое предсказание по «хвостам»). Парадигма найденной словоформы используется как образец для создания парадигмы входной словоформы. Необходимо отметить, что в анализаторе Информэлектро модель управления неопознанных словоформ также предсказывается по «хвостам».

Все три рассмотренные системы морфологий с использованием лексиконов демонстрируют сравнимые по скорости и точности анализа результаты.

III. Метод снятия морфологической омонимии (tagger).

Еще в начале 60-ых годов американский лингвист Ч. Хоккеттом указал на возможность использования конечной марковской цепи в качестве модели для описания процесса синтаксического анализа, возникающего в голове слушающего после восприятия каждого последующего слова, произнесенного говорящим, в предложении [Ч. Хоккетт, 1961]. В компьютерной лингвистике скрытые марковские модели нашли свое применение в задачах разрешения омонимии словоформы по синтаксическому контексту в предложении.

Входными данными модуля tagger в LxPlatform служат результаты графематического и морфологического анализов, полученных модулями tokenizer и stemmer. Tagger представляет собой скрытую марковскую модель, способную запоминать последовательности длиной от 4 до 6 синтаксических единиц. Коэффициенты вероятностей выбора морфологических значений вырабатываются в цепи путем обучения марковской модели на размеченном тексте. Каждой словоформе в размеченном тексте присваивается морфологическая помета (tag). Для того, чтобы сократить размеры как самой скрытой модели, так и размеченного текста, необходимого для обучения, используются усеченные морфологические пометы, которые позволяют сократить комбинаторно возможные варианты синтаксических контекстов. Так, полная морфологическая помета словоформы ‘красивому’ ‘+Adj+Plain+Sg+MascNeut+Dat’ будет усечена в tagger до пометы ‘Adj-Obl’, такую же помету получат и другие формы прилагательного ‘красивый’, стоящие в косвенных падежах. Все финитные формы глагола используют единую помету Verb-Fin. В таблице перечислены все морфологические пометы, составляющие алфавит марковской модели для русского языка:

Помета	Описание	Примеры

Adj-Nom	Прилагательное в номинативе	красивый, красивая, красивое, красивые
Adj-Acc	Прилагательное в аккузативе	красивого, красивую, красивое, красивые
Adj-Gen	Прилагательное в генитиве	красивого, красивой, красивых
Adj-Obl	Прилагательное в косвенном падеже	красивым, красивой, красивому, красивыми
Adj-Comp	Сравнительная степень прил.	краше
Adj-Brf	Краткая форма прил.	красив, красива, красиво, красивы
Adv	Наречие	быстро
Conj	Союз	и, но, чтобы
Det-Nom	Местоименное прил. в номинативе	этот
Det-Acc	Местоименное прил. в аккузативе	эту
Det-Gen	Местоименное прил. в генитиве	этого
Det-Obl	Местоименное прил. в косвенном падеже	этому
Dig	Цифровой комплекс	1999, 100Мб
Pron-IntRel-Nom	Относительные местоимения в номинативе	кто
Pron-IntRel-Acc	Относительные местоимения в аккузативе	кого, что
Pron-IntRel-Gen	Относительные местоимения в генитиве	кого, чего
Pron-IntRel-Obl	Относительные местоимения в косвенном падеже	кому
Interj	Междометие	ага, ах, ба
Nn-Nom	Существительное в номинативе	сестра, сестры
Nn-Acc	Существительное в аккузативе	сестру, сестер
Nn-Gen	Существительное в генитиве	сестры, сестер
Nn-Obl	Существительное в косвенном падеже	сестрой, сестрами
Num	Числительное	три, восемь
Ord	Цифра	7., 3.
Pron-Pers-Nom	Личное местоимение в номинативе	я, ты
Pron-Pers-Acc	Лич. местоим. в аккузативе	меня, тебя
Pron-Pers-Gen	Лич. местоим. в генитиве	меня, тебя
Pron-Pers-Obl	Лич. местоим. в косвенном падеже	мной, тобой
Prep-Nom	Управляющий номинативом предлог	плюс, минус
Prep-Acc	Управляющий аккузативом предлог	за
Prep-Gen	Управляющий генитивом предлог	без, накануне
Prep-Obl	Управляющий косвенным падежом предлог	благодаря, к
Pron-Nom	Местоимение в номинативе	все, ничто
Pron-Acc	Местоимение в аккузативе	все, ничто
Pron-Gen	Местоимение в генитиве	всего, ничего
Pron-Obl	Местоимение в косвенном падеже	всеми, ничем
Prop-Nom	Имя собственное в номинативе	Москва, Мальцев
Prop-Acc	Имя собственное в номинативе	Москву, Мальцева
Prop-Gen	Имя собственное в генитиве	Москвы, Мальцева
Prop-Obl	Имя собственное в косвенном падеже	Москве, Мальцеве
Part	Частица	аж, же
Part-Int	Вводное	авось, конечно
Part-Sent	Предикатив	аминь
Aux	Вспомогательный глагол	быть
Verb-Fin	Финитная форма глагола	делай, делает, делал

Verb-Ger	Деепричастие	делав, делавши, делая
Verb-Inf	Инфинитив	делать
Verb-Acc	Причастие в аккузативе	делавшего, делавшее, делавшую
Verb-Gen	Причастие в генитиве	делавшего, делавшей
Verb-Nom	Причастие в номинативе	делавший, делавшее, делавшая
Verb-Obl	Причастие в косвенном падеже	делавшим, делавшей
Verb-Brf	Краткое причастие	делан, делано, делана

С уменьшением числа морфологических помет понижается и точность синтаксического контекста, а вместе с ним и анализа. Такая вероятностно-статистическая модель, учитывающая синтаксический контекст, косвенно лишена проверки полного согласования. Но экспериментальные данные доказывают, что даже такого числа усеченных помет достаточно для 95% точности при выборе леммы и грамматического значения словоформы, т.е. минимальный объем модели позволяет с высокой точностью снимать морфологическую омонимию. Действительно, обучение скрытой марковской модели на размеченном приведенными в таблице пометами тексте, размер которого не превышает 300 Кб, позволяет вычислять ожидаемые вероятностные коэффициенты для выбора правильного грамматического значения в простых и частотных случаях контекстного распределения.

Приведем результаты анализа модулями stemmer и tagger двух пар предложений, содержащих омонимичные словоформы, принимающие разные леммы и грамматические значения в зависимости от контекстного распределения.

Исходный текст:

На завод привезли стекло.
Масло стекло на пол.

Данные эксперименты являются ошибочными.
Последние данные являются ошибочными.

Результат лемматизации stemmer:

```
На          на
завод       завод
привезли    привозить
стекло      стекло | стекать
.           .
Масло       масло
стекло      стекло | стекать
на          на
пол         пол | пола | полый
.           .
Данные      давать | данные | данный
эксперименты эксперимент
являются    являть | являться
ошибочными  ошибочный
.           .
Последние   последние | последний
данные      давать | данные | данный
являются    являть | являться
ошибочными  ошибочный
.           .
```

Результат выбора значений tagger:

На [Prep-Acc] на
завод [Nn-Acc] завод
привезли [Verb-Fin] привозить
стекло [Nn-Acc] стекло
. [Punct-Sent] .
Масло [Nn-Nom] масло
стекло [Verb-Fin] стекать
на [Prep-Acc] на
пол [Nn-Acc] пол
. [Punct-Sent] .
Данные [Adj-Nom] данный
эксперименты [Nn-Nom] эксперимент
являются [Verb-Fin] являть | являться
ошибочными [Adj-Obl] ошибочный
. [Punct-Sent] .
Последние [Adj-Nom] последний
данные [Nn-Nom] данные
являются [Verb-Fin] являть | являться
ошибочными [Adj-Obl] ошибочный
. [Punct-Sent] .

Метод снятия омонимии, основанный на скрытой марковской цепи, - редкий случай, когда вероятностно-статистическая модель эффективно работает в лингвистике.

IV. Методика выделения именных групп (np-grouper).

Язык регулярных выражений – формальный язык, во многом схожий с формулами булевой логики. Он обладает простым синтаксисом, но выражения могут быть произвольно сложными. Каждое выражение обозначает множество. Позволяя создавать гибкие образцы (шаблоны) для любых последовательностей элементов, язык регулярных выражений широко применяется для быстрого поиска подстрок и обработки нечетких запросов. Регулярные выражения компилируются в конечные автоматы, что позволяет достигать высокой скорости при поиске шаблонов.

Модуль np-grouper в LxPlatform предназначен для выделения из предложений именных составляющих NP. Фактически, np-grouper можно считать начальным этапом синтаксического анализа предложения. Такая технология используется в задачах автоматической обработки текстов (автоматическое построении таксономии и классификация информационного потока) с последующим статистическим анализом найденных NP. Для создания образцов NP, последовательностей элементов внутри группы, используется язык регулярных выражений, где каждое выражение представляет собой грамматический образ некоторой именной группы или ее подгруппы. Множество таких выражений составляет грамматику именных групп.

Регулярное выражение заключено в квадратные скобки ‘[…]’. Определение ‘**define** *name* […] ‘ присваивает уникальное имя выражению. В круглые скобки ‘(…)’ заключается факультативная последовательность элементов внутри выражения. Символ ‘?’ означает

любой (any) символ. Унарные операторы языка: ‘*’ – звезда Клини; ‘+’ – плюс Клини. Бинарные операторы языка: пробел между двумя элементами означает их конкатенацию; ‘|’ означает дизъюнкцию.

Построим грамматику именных групп для русского, используя морфологические пометы, введенные в предыдущем разделе, которые приобретают характер синтаксических элементов в нашей грамматике.

```
# определим подгруппы полных прилагательных и причастий, модифицированных наречием

define ANOM [Adj%-Nom | Verb%-Nom];
define AACC [Adj%-Acc | Verb%-Acc];
define AGEN [Adj%-Gen | Verb%-Gen];
define AOBL [Adj%-Obl | Verb%-Obl];

define AdjPNom [ ANOM+ [[Conj | Punct%-Comma] (Adv) ANOM]* ];
define AdjPAcc [ AACC+ [[Conj | Punct%-Comma] (Adv) AACC]* ];
define AdjPGen [ AGEN+ [[Conj | Punct%-Comma] (Adv) AGEN]* ];
define AdjPObl [ AOBL+ [[Conj | Punct%-Comma] (Adv) AOBL]* ];

# объединим существительные и имена собственные для каждого падежа соответственно

define NNNOM [Nn%-Nom | Prop%-Nom];
define NNACC [Nn%-Acc | Prop%-Acc];
define NNGEN [Nn%-Gen | Prop%-Gen];
define NNOBL [Nn%-Obl | Prop%-Obl];

# определим числительные

define NUMBER [Num+ | Dig];

# определим пре-модифицированные именные группы

define ANPNom [ (AdjPNom) NNNOM ];
define ANPAcc [ (AdjPAcc) NNACC ];
define ANPGen [ (NUMBER (Adv)) (AdjPGen) NNGEN ];
define ANPGen1 [ (Det%-Gen) (Adv) (AdjPGen) NNGEN ];
define ANPObl [ (AdjPObl) NNOBL ];

# определим однородные именные группы

# определим однородные пары именных групп, соединенных сочинительным союзом

define HOMOGENPairNom [ ANPNom Conj (Adv) ANPNom ANPGen1* ];
define HOMOGENPairAcc [ ANPAcc Conj (Adv) ANPAcc ANPGen1* ];
define HOMOGENPairGen [ ANPGen Conj ANPGen1+ ];
define HOMOGENPairObl [ ANPObl Conj (Adv) ANPObl ANPGen1* ];

# определим однородные цепочки именных групп длиной от 4 элементов и больше.
# определение трехсоставных однородных цепочек может повлечь
# высокий процент ошибочно построенных однородных групп

define HOMOGENNom [ANPNom [Punct%-Comma (Adv) ANPNom]+ Punct%-Comma (Adv) HOMOGENPairNom];
define HOMOGENAcc [ANPAcc [Punct%-Comma (Adv) ANPAcc]+ Punct%-Comma (Adv) HOMOGENPairAcc];
define HOMOGENGen [ ANPGen [Punct%-Comma ANPGen1]+ Punct%-Comma (Adv) HOMOGENPairGen];
define HOMOGENObl [ ANPObl [Punct%-Comma (Adv) ANPObl]+ Punct%-Comma (Adv) HOMOGENPairObl];

# определим пост-модификацию именной группы существительным в генитиве
```

```
define NNS [ ANPNom | ANPAcc | ANPGen | ANPObl ];

define NNS2 [ [(AdjPNom) Nn%-Nom] | [(AdjPAcc) Nn%-Acc] | [(NUMBER (Adv)) (AdjPGen) Nn%-Gen] | [(AdjPObl) Nn%-Obl] ];

define NPGENIT [ NNS2 ANPGen1+ (Conj ANPGen1) ];

define NUMBERNP [ NUMBER Nn%-Gen NUMBER Nn%-Gen ];

# определим пост-модификацию именной группы прилагательным

define ANPPOSTMNom [ NNNOM Adj%-Nom ];
define ANPPOSTMAcc [ NNACC Adj%-Acc ];
define ANPPOSTMObl [ NNOBL Adj%-Obl ];

# определим именную группу состоящую из цепочки существительных и имен собственных

define NPPROPNOM [ NNNOM [Prop%-Nom]+ ];
define NPPROPACC [ NNACC [Prop%-Acc]+ ];
define NPPROPGEN [ NNGEN [Prop%-Gen]+ ];
define NPPROPOBL [ NNOBL [Prop%-Obl]+ ];

# определим разные классы NP

define NPHOMOGENPAIR [ HOMOGENPairNom | HOMOGENPairAcc | HOMOGENPairGen | HOMOGENPairObl ];

define NPHOMOGEN [ HOMOGENNom | HOMOGENAcc | HOMOGENGen | HOMOGENObl ];

define NPPOSTM [ ANPPOSTMNom | ANPPOSTMAcc | ANPPOSTMObl ];

define NPPROP [ NPPROPNOM | NPPROPACC | NPPROPGEN | NPPROPOBL ];

# определим NP образец

define NPS [
NNS |
NPHOMOGENPAIR |
NPHOMOGEN |
NPGENIT |
NUMBERNP |
NPPOSTM |
NPPROP
];
```

Таким образом, такая грамматика именных групп способна определять именные составляющие NP следующих типов: (а) существительные, пре-модифицированные прилагательным, причастием или числительным (‘очень разумная идея’, ‘белый, красный и зеленый шар’, ‘два стола’); (б) сочиненные именные группы (‘брат и сестра’, ‘низкий стол, стул, широкий табурет и шкаф’); (в) генитивные группы (‘рука власти’, ‘Министерство Финансов’); (г) цепочки существительных (‘город Москва’, ‘Сергей Петрович Иванов’); (д) определение в постпозиции, выраженное прилагательным (‘впечатление необычное’).

Основным недостатком такого формализма является невозможность описания разрывных составляющих на языке регулярных выражений. Усеченные морфологические

пометы лишают возможности проверки полного согласования, оставляя только частичное падежное согласование в грамматике. Подобная модель шаблонов именных групп не способна выделять два типа NP, определенных для русского языка: необособленное согласованное определение и пост-модификация именной группы, выраженная предложной группой.

Экспериментальные данные и проведенное тестирование модуля np-grouper доказывает работоспособность методики и относительно высокую точность (не менее 98%) построения NP. Достоинством грамматики именных групп, сформулированной на языке регулярных выражений, является ее краткость и прозрачность.

Все приведенные в настоящей главе морфологические и предсинтаксические компоненты анализа потенциально являются неотъемлемой частью идеальной модели полного синтаксического процессора, а также позволяют демонстрировать общность формализма и методов решения задач на разных уровнях лингвистического анализа.