Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение высшего образования
«Московский государственный технический университет имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ «ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ»

КАФЕДРА «СИСТЕМЫ ОБРАБОТКИ ИНФОРМАЦИИ И УПРАВЛЕНИЯ»

ОТЧЕТ ПО ЭКСПЛУАТАЦИОННОЙ ПРАКТИКЕ

Студента Фадеев Артем Александрович, ИУ5-21М
фамилия, имя, отчество, группа

Тип практики Учебная

Место практики НУК ИУ МГТУ им. Н.Э. Баумана

Студент _____ **Фадеев А.А.**
подпись, дата *фамилия, и.о.*

Руководитель практики _____ **Варламов О.О.**
подпись, дата *фамилия, и.о.*

Оценка _____

2021 г.

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение высшего образования
«Московский государственный технический университет имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

З А Д А Н И Е

по эксплуатационной практике

Магистранту: Фадеев А.А.

Группа: ИУ5-21М

Вид практики: Учебная

Срок практики: с 01.07.2021 по 21.07.2021

Место практики: НУК ИУ МГТУ им. Н.Э. Баумана

Руководитель практики: Варламов О.О.

1. Индивидуальное задание

Исследовать технологии создания синтаксических анализаторов для формирования метаграфовой базы знаний на основе русского текста

2. План прохождения практики

№ п/п	Вид работы, форма отчетности	Срок выполнения	Отметка о выполнении
1	Проведение научной работы по заданной теме исследования, отчет по практикуму	01.07.2021 – 21.07.2021	

Студенты _____ (Фадеев А.А.)

Руководитель практики _____ (Варламов О.О.)

Примечание: Задание оформляется в двух экземплярах: один выдается студентам, второй хранится на кафедре.

Оглавление

Аннотация 4

Введение 4

Описание концепта 5

Формирование структуры правил 11

Укрупненная схема базы данных 16

Заключение 16

Список литературы 17

Аннотация

Сейчас уровень распознавания русско-язычного текста в среднем находится на уровне 80-85 процентов. При этом компьютер не в состоянии понять текст, следовательно, извлечь знания из него. В результате этого велика вероятность, что со временем многие знания потеряются, т.к. уже сейчас человек не может обработать весь объём знаний, который сумел накопить. То же касается и знаний на каждом отдельном языке, в частности, на русском языке. Метаграфовая модель знаний позволит легко оперировать информацией, элементы которой тесно переплетены с помощью информативных связей. Благодаря этому повысится потенциальная сложность исследуемого текста и точность его понимания программой. Автомат сможет находить противоречия в различных блоках знаний, связывать имеющиеся исследования и надёжно сохранять уже имеющиеся знания.

Введение

Квалификационная работа на тему «Парсер ЕЯ в метаграфовую модель для русского языка» посвящена разработке модуля для построения метаграфа по результатам синтаксического анализа исходного русскоязычного текста.

В данной области множество научных работ и различных продуктов, дошедших до простого пользователя: электронные переводчики, распознаватели текста, поисковые системы, вопросно-ответные системы. Но почти все доступные сторонним пользователям системы не понимают слова так, как это делает человек, а лишь обрабатывают цепочки символов, используя статистику.

При работе с ЕЯ основными задачами являются разбор и анализ входного текста или синтез выходных предложений. Анализа текстов на ЕЯ состоит из 6 этапов: разделительный (токенизация), морфологический, фрагментационный, синтаксический, семантический и прагматический.

В данной квалификационной работе подробно рассматривается синтаксический этап разбора входного текста. Токенизация и морфологический этап разбора проводится за счёт сторонних программных средств. Фрагментационный анализ текста не рассматривается, однако для его реализации также используется программа собственной разработки.

Морфологический анализ включает определение морфологических характеристик слова, таких как часть речи, род, число, падеж и т.д.

Синтаксический анализ, который содержит определение синтаксических ролей слов и словосочетаний в предложении. Результатом анализа является метаграф синтаксических зависимостей слов, который отражает структуру их взаимодействия в рамках одного предложения.

Описание концепта

Парсер позволит получать легко интерпретируемую метаграфовую модель морфологически-синтаксических знаний по тексту. Парсер позволит производить синтаксический анализ независимо от семантического, что снизит вычислительную сложность алгоритма полноценного анализа текста. Конечный результат данной работы будет использован для разработки прототипа метаграфовой базы знаний на основе русскоязычной литературы. Данный прототип ориентирован на анализ научных текстов для сохранения актуальных знаний. Разрабатываемые технологии анализа текста смогут найти применение при создании анализаторов текстов на любом языке. В частности, при создании программ-переводчиков, бытовых и промышленных роботов-помощников, которые будут способны взаимодействовать с людьми, полноценно владея естественным языком.

Для построения архитектуры синтаксического анализатора необходимо проанализировать все решения, которые используются в разработке синтаксических анализаторов.

Детальный анализ приведён в трудах [1] и [2], а также работе [9]. В них говорится о различных стратегиях при анализе синтаксических конструкций с разной степенью детализации.

Наиболее четкие границы есть между тремя подходами к описанию синтаксиса:

1) Грамматики зависимостей;

2) Грамматики непосредственных составляющих;

3) Комбинированные теории, например, теория синтаксических групп.

Основной принцип подхода грамматики зависимостей заключается в том, что каждое слово предложения составляет иерархию. Каждое слово является своего рода атомом в молекуле. Это означает что само предложение в результате анализа представляется в виде дерева зависимостей или, другими словами, графа зависимостей. Таким образом синтаксические связи устанавливают между словами отношения зависимости. Одно из двух слов является главным, а другое зависимым. Например, в словосочетании «Большая Советская Энциклопедия» имеется две связи, изображённые на рисунке ниже (см. рис.1). Эти связи образуют отношения зависимости: в обеих связях главной является словоформа энциклопедия. Словоформы «Большая» и «Советская» оказываются зависимыми.

Рис. 1. Строение взаимосвязей в «грамматике зависимостей»

Как мы видим, узким местом этого подхода является невозможность построения какой-либо другой связи, кроме подчинительной.

Непосредственные составляющие (НС, фразы) – на рисунке ниже (см. рис.2) приведён пример структуры предложения "Мама мыла раму". У этого предложения имеются НС: "мама" и "мыла раму". В свою очередь, "мыла раму" имеет составляющие "мыла" и "раму", то есть эти НС не входят непосредственно в предложение, а являются компонентами его более-менее крупной части. По этой схеме ни одна словоформа не будет являться НС целого предложения, но будет являться НС по отношению к тем или иным его составляющим.

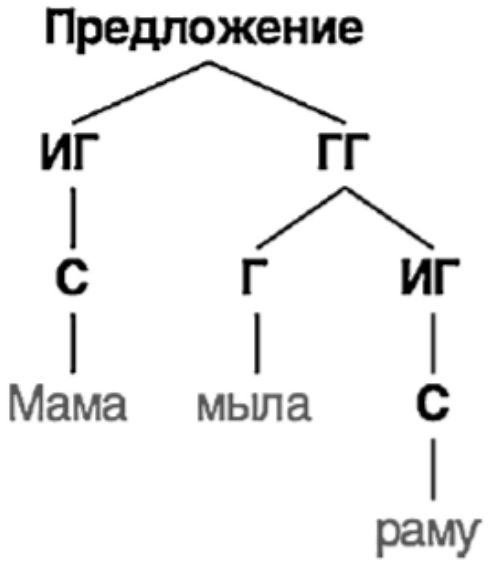

Рис. 2. Грамматика непосредственных составляющих.

По данной схеме предложение состоит из одной или нескольких НС. Каким образом данные НС-грамматика оперирует такими понятиями, как классы НС, например, предложения, именная группа или глагольное группа. Однако названия этих классов никак не обосновываются частями речи тех словоформ, которые входят в его состав.

Именно в подходе непосредственных составляющих появляется такое понятие как восходящие алгоритмы или нисходящие алгоритмы. Подробное описание восходящих и нисходящих алгоритмов описано в работе [2]. Для данной работы «восходящий» способ был выбран именно потому, что он отличается относительной простотой в плане построения правил нахождения взаимосвязи между различными НС. Основной принцип такого вида алгоритма заключается в том, что в предложении ищется наименее значимая с синтаксической точки зрения словоформа, и она соединяется с более значимой по специальным правилам, например, "частица+глагол" или "предлог+существительное". Таким образом мы получаем возможность отсечения неверных вариантов на том этапе разбора предложения, когда мы не обработали все словоформы и не порождаем множество заведомо неверных вариантов. Таким образом мы можем экономить время работы и ресурсы компьютера. Одним из основных минусов такого вида алгоритмов является порождение так называемых эллипсисов или рекурсивных грамматик. Эллипсис – возникновение пропусков некоторых слов в предложении. В таком случае мы должны восстанавливать потерянные слова из контекста. Примеры предложений с эллипсисом: "Он нравится Маше, а она – ему".

Комбинированные алгоритмы нивелируют минусы друг друга за счёт того, что в них может реализовываться разный подход к использованию восходящего алгоритма и нисходящего алгоритма. Например, сначала может быть разобранной восходящем алгоритмом именная группа подлежащего в предложении, а нисходящем – группа сказуемого. За счет использования принципов обоих подходов комбинированные алгоритмы находятся по

середине между восходящими и нисходящими алгоритмами в плане производительности и скорости. Однако они впереди нисходящих и восходящих алгоритмов в плане точности разбора предложений, поскольку они могут реализовывать разбор эллипсисов, рекурсивных грамматик или других сложных случаев предложений, которые недоступны для разбора обычным восходящим или нисходящим алгоритмам.

На рисунке 3 изображены общие схемы работы трех описанных выше алгоритмов. Число рядом с НС показывает порядок нахождения зависимостей.

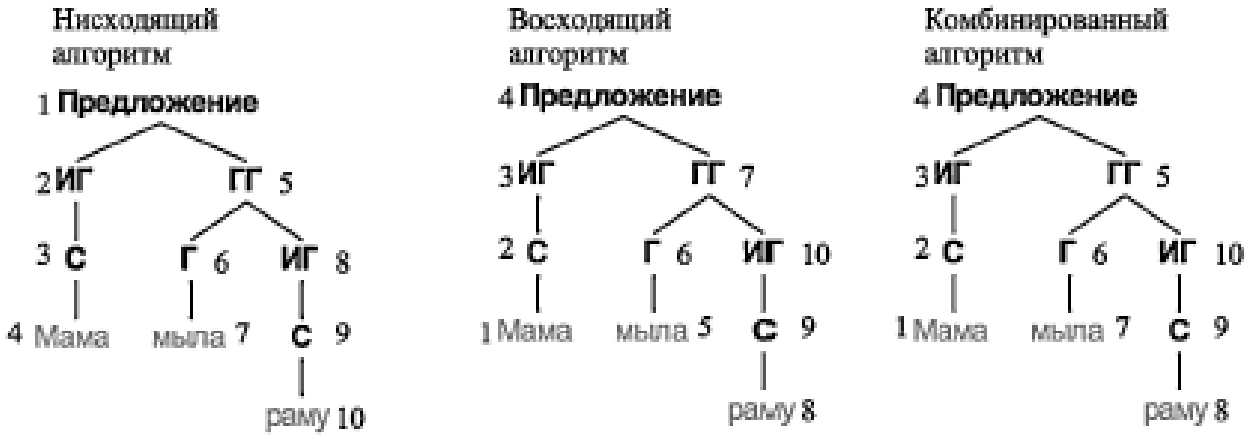

Рисунок 3. Порядок работы нисходящих, восходящих и комбинированных алгоритмов

Еще одним минусом восходящих и нисходящих алгоритмов НС является их неспособность анализировать те НС, которые разделены другими НС. Для устранения недостатка подобных алгоритмов А. В. Гладким была разработана теория синтаксических групп (ТСГ), которая реализуется в разрабатываемой системе.

Этот подход является представителем комбинированных теорий анализа предложения [2]. Он предполагает использование восходящего алгоритма, но уже применительно не к непосредственным составляющим, а к синтаксическим группам (СГ). ТСГ допускает включение в структуру предложения тех групп, которые участвуют в отношении зависимости

«целиком», а не посредством одной словоформы. Таким образом этот подход позволяет легко обрабатывать разрывные составляющие.

Независимо от выбранного подхода главная проблема синтаксического анализа связана с неоднозначностью языковых единиц. Помимо очевидной многозначности слов (лексической неоднозначности), есть и менее очевидная многозначность грамматических форм (морфологическая неоднозначность): ср., например, например, числи тельное три и форму повелительного наклонения единственного числа глагола тереть, или форму именительного падежа слова стекло и форму прошедшего времени, единственного числа, среднего рода глагола стекать. Морфологическая неоднозначность может показаться редким явлением, но только потому, что она редко бывает очевидной.

Больше всего проблем вызывает, однако, не морфологическая, а синтаксическая неоднозначность. Из-за морфологической неоднозначности одна и та же фраза может иметь несколько разных пониманий, которые соответствуют разным структурам. Более того, одна и та же фраза может быть понята по-разному даже при однозначности всех её словоформ («Вася встретил Петю в коридоре» и «Вася встретил Петю в костюме»). В действительности, обе фразы имеют, по крайней мере, три возможные синтаксических структуры. Им соответствует три разные ситуации. Для первой фразы они таковы:

– Вася встретил Петю, и это произошло в коридоре;
– Вася встретил Петю, одетого в коридор;
– Вася, одетый в коридор, встретил Петю.

Чтобы найти верный вариант разбора целого предложения, можно попытаться перебрать все комбинации вариантов разбора его частей. Проблема в том, что при компьютерном анализе это приводит к так называемому комбинаторному взрыву.

Как уже видно из примера даже у самого короткого предложения может быть несколько интерпретаций на уровне синтаксиса и лексики. Поэтому, если стоит цель выдать единственно верный вариант синтаксического разбора, необходимо решить задачу, по сложности сопоставимую с задачей коммивояжера. Однако поиск верного решения среди нескольких вариантов – задача семантического характера, поэтому ее решение идет в разрез с целью данного проекта.

Выбранный алгоритм СА предполагает, что каждая СГ части предложения равноправна любой другой. Исключения составляют слова грамматической основы. Если слово – часть грамматической основы, то оно не может быть подчинено, а слова, подчиняемые словам грамматической основы, присоединяются с ней на предпоследнем шаге. Таким образом любое слово, кроме грамматической основы в теории может быть связано с любым другим словом предложения. Само предложение считается проанализированным, когда каждое слово, кроме грамматической основы стало подчиненным. Тогда грамматическая основа связывается в предложение одним из трёх уникальных правил: только «Подлежащее», только «Сказуемое» и «Подлежащее+Сказуемое или Сказуемое+Подлежащее».

Формирование структуры правил

В связи с выбранным методом синтаксического анализа, правила должны определять взаимодействие двух синтаксических групп без участия семантики текста. Таким образом, предложенные в работах [4] и [5] системы правил не подходят разрабатываемой системе. Так как алгоритм анализа можно охарактеризовать как восходящий, каждое правило имеет приоритет применения. Правила хранятся в БД, схема которой описывается в пункте 2.6.

Для работы правил необходима информация о двух СГ, составляющих объекты применения правила, о том, какая из двух СГ – главная, а какая –

подчиненная. Также необходима информация о самом приоритете правила и о флагах согласования двух СГ по роду, числу и падежу.

Основной задачей в процессе изучения предметной области является выявления зависимостей и закономерностей формирования синтаксических конструкций.

В основе синтаксического анализа выбранного подхода лежит построение синтаксических групп на одном морфологическом варианте разбора части предложения. Частью называется простое предложение, которое содержит только одну грамматическую основу. Перед непосредственно анализом одного фрагмента производится фрагментация сложного предложения на простые.

Существует два типа синтаксического анализа:

– Нисходящий синтаксический анализ. При нисходящем анализе делается предположение о строении предложения и проверяется путем сравнения с исходным. Таким образом, движение идет от корня к листьям.

– Восходящий синтаксический анализ. При восходящем синтаксическом анализе с помощью набора правил единицы предложения объединяются в группы, пока в предложении не останется одна группа, которая будет соответствовать корню – грамматической основе.

Для того, чтобы унифицировать применение различных правил необходимо заранее разработать структуру, которая будет учитывать все возможные варианты и разработать алгоритм применения правил, соответствующих данной структуре. Слово рассматривается исключительно как СГ, и не важно, сколько слов она подчинила и подчинила ли. Каждое словосочетание (СГ) выполняет определенную синтаксическую функцию, которая определяется главным словом в данном словосочетании. Словосочетания могут объединятся в более крупные, сохраняя роль главного или приобретая новую.

Синтаксическая группа (СГ) может обладать синтаксической ролью, также в ней определено главное слово, которое определяет роль словосочетания. В таком случае и слова, и словосочетания будут является синтаксическими группами.

На вход алгоритма подаются две синтаксические группы, по которым делается запрос в таблицу БД «Правило». Алгоритм определяет уникальное правило сочетания именно этих двух СГ. Далее по приоритету выбранного правила определяется, необходимо ли применять данное правило (подробное описание алгоритма анализа с применением правил приводится в пункте 2.7.2). Затем правило применяется, и из двух СГ остается доступной для анализа только одна из них: главная.

Всего у входных СГ может быть два типа: главная и подчиненная.

В русском языке есть правила сочетания слов, как правило, не закрепленные документально. В русском языке нет правил порядка следования членов предложения в связи с чем моментально и многократно возрастает сложность нахождения отношений между словами. Однако с этим ничего не поделать, поэтому необходимо просто обработать в два-три раза больше вариантов сочетаний слов, чем для других языков.

Отношения СГ, по сути, отношения зависимости с некоторыми оговорками, поскольку возможно нахождение однородных членов и даже однородных простых предложений в составе сложного. Однако правила устанавливают именно отношения зависимости, выстраивая иерархию слов предложения.

Само правило хранит свой приоритет для определения порядка применения правил. Также правило содержит информации о необходимом согласовании входных групп, которые к нему относятся. Это позволяет обеспечивать требования согласования в различных правилах по роду, числу и падежу.

Все правила однозначно определяют итоговый член предложения, который реализует подчиненная СГ данного словосочетания.

Как говорилось ранее у каждого правила есть приоритет. Этот приоритет соответствует порядку построения групп: от меньших к большим. Например, сначала надо построить группы нареч-ПРИЛ, а потом прил-СУЩ, чтобы построить структуру на отрезке "очень красивый человек": прил-СУЩ (нареч-ПРИЛ (очень красивый), человек) Каждое правило применяется к каждому слову входного отрезка слева направо.

Ниже представлены примеры правил по каждому из 9 приоритетов и их описания в рамках входных синтаксических групп. Также здесь приведено правило, объединяющее грамматическую основу в предложение.

Чем больше номер приоритета, тем больше вероятность нарушения связей предложения при простановке для этого правила меньшего номера. Например, есть словосочетание «за чем-то очень приятные слова». Это сочетание «Предлог + Местоимение + Наречие + Прилагательное + Существительное». Мы понимаем, что возможны сочетания «Местоимение + Наречие», однако это менее вероятное сочетание, чем присутствующее в данном примере «Местоимение + Прилагательное» и «Наречие + Прилагательное», поэтому его приоритет становится больше, чем приоритет того же «Наречие + Прилагательное».

Также мы видим, что Предлог стоит через три слова от своего главного – Существительного, поэтому сочетания предлога и наречия менее приоритетны, предлога и местоимения или прилагательного более приоритетны, а предлога и существительного – самые приоритетные и имеют приоритет, равный 1.

Так как разбор идет слева направо, не возникнет конфликтных ситуаций внутри цепочки СГ, где главная группа следует за подчиненной. Во всех

остальных ситуациях конфликт может возникнуть и эти потенциальные конфликты решаются именно ведением согласования и приоритетов.

Правила построены так, чтобы любое слово, даже грамматическая основа, сохранилось в БД в таблице «Подчиненная СГ». Таким образом предложение станет восстановимым в своем изначальном виде благодаря лишь данным в одной таблице БД.

На рисунке 4 приведена структура правила «предл+СУЩ».

Правила для построения групп
«Предложная группа» + «Существительная группа»

Главная группа: Существительная группа;

Подчиненная группа: Предложная группа;

Приоритет: 1;

Что объединяет: одиночные предлог и существительное;

Подчинённый член предложения: Служебный;

Примеры: *в машине, на балконе, под окном*.

Рисунок 4. Структура правила «предл+СУЩ»

Согласование по падежу, числу и роду не нужно, поскольку Предлог не имеет этих характеристик.

Это правило имеет первый приоритет. Это означает, что оно обработается первым среди всех прочих.

Существительная группа – главная, поэтому после применения правила она останется в предложении, а группа предлога станет считаться обработанной и в дальнейшем поиске двух СГ для применения правила использоваться не будет. Поэтому после применения правила данная связка

слов станет СГ вида Существительная группа, а предлог станет Служебным членом предложения.

Укрупненная схема базы данных

В данном проекте база данных отвечает исключительно за хранение данных о правилах, и необходима лишь для того, чтобы хранить информацию о разобранных словах предложения. То есть каждый успешный результат анализа предложения записывается в Базу Данных.

– Сущность Правило – хранит данные обо всех правилах Синтаксического Анализа (СА);

– Сущность Вид синтаксической группы (СГ) – входная характеристика для применения правила (правила применяются к двум СГ, составляя новую СГ);

– Сущность Тип синтаксической группы – входная характеристика для применения правила;

– Сущность Граммема – хранит данные о любой возможной грамматической характеристике лексемы (род, падеж и т.п.);

– Сущность Главная синтаксическая группа – хранит результирующие СГ, являющиеся главными в предложении, включая само предложение, подчиняющее подлежащее и сказуемое;

– Сущность Подчиненная синтаксическая группа – хранит подчиненные СГ.

Заключение

Были рассмотрены подходы и введены новшества к проведению синтаксического анализа, построению метаграфа по результатам синтаксического анализа.

Список литературы

1. Волкова, И. А. Введение в компьютерную лингвистику. Практические аспекты создания лингвистических процессоров: Учебное пособие для студентов ВМиК МГУ / И. А. Волкова – Москва: Издательский отдел факультета ВМиК МГУ, 2006. – 43 с.

2. Прикладная и компьютерная лингвистика / ред. Николаев И. С., Митренина О. В., Ландо Т. М. – 2-е изд., – Москва: Издательская группа URSS, 2017. – 320 с.

3. Гапанюк, Ю. Е. Конспект лекций по спецкурсу «Гибридные интеллектуальные информационные системы на основе метаграфового подхода» / Ю. Е. Гапанюк – Москва: Издательство «Спутник +», 2018. – 56 с.

4. Тузов, В. А. Компьютерная семантика русского языка / В. А. Тузов – СПб.: Изд-во СПбГУ, 2003. – 391 с.

5. Мартынов, В. В. Основы семантического кодирования. Опыт Представления и преобразования знаний / В. В. Мартынов. – Минск: ЕГУ, 2001 г. – 140 с.

6. Проект «Открытый корпус» (OpenCorpora) [Электронный ресурс] – 2009-2020 г. – Режим доступа: http://opencorpora.org/dict.php?act=gram, свободный.

7. Морфологический анализатор pymorphy2 [Электронный ресурс] / М. Коробов – 2015-2020 г. – Режим доступа: https://pymorphy2.readthedocs.io/en/latest/index.html, свободный.

8. Постников, В. М. Основы эксплуатации АСОИиУ Том 1. Техническое обслуживание АСОИиУ. 2-е издание, переработанное и дополненное, учеб. пособие / В. М. Постников. – М.: Изд-во МГТУ им. Н. Э. Баумана, 2015. – 214 [2] с.

9. Леонтьева, Н. Н. Автоматическое понимание текстов: системы, модели, ресурсы: учеб. пособие для студ. лингв. фак. вузов / Нина Николаевна Леонтьева. – М.: Издательский центр «Академия», 2006. – 304 с.

10. ГОСТ 19.201-78. ЕСПД. Техническое задание. Требования к содержанию и оформлению.

11. ГОСТ 19.404-79. ЕСПД. Пояснительная записка. Требования к содержанию и оформлению.

12. ГОСТ 19.301-79. ЕСПД. Программа и методика испытаний. Требования к содержанию и оформлению.

13. Зализняк, А. А. Грамматический словарь русского языка: Словоизменение. 3-е изд. / М.: Изд-во «Русский язык», 1987. 880 с.

14. Хворостин, Д. В. Англо-Русский словарь лингвистических терминов / Д. В. Хворостин – Челябинск, 2007 г. – 113 с.

15. Парсер Solarix – морфологический и синтаксический анализатор русскоязычных текстов [Электронный ресурс] / Илья Козиев – 2019 г. – Режим доступа: http://www.solarix.ru, свободный.

16. Автоматическая обработка текста АОТ [Электронный ресурс] – Режим доступа: http://aot.ru/docs/synan.html/, свободный.