Федеральное государственное бюджетное учреждение науки
Институт системного анализа
Российской академии наук

На правах рукописи

Шелманов Артем Олегович

ИССЛЕДОВАНИЕ МЕТОДОВ АВТОМАТИЧЕСКОГО АНАЛИЗА ТЕКСТОВ И РАЗРАБОТКА ИНТЕГРИРОВАННОЙ СИСТЕМЫ СЕМАНТИКО-СИНТАКСИЧЕСКОГО АНАЛИЗА

Специальность: 05.13.17 – Теоретические основы информатики

Диссертация на соискание ученой степени
кандидата технических наук

Научный руководитель –
доктор физико-математических наук,
профессор Г.С. Осипов

Москва 2015

СОДЕРЖАНИЕ

ВВЕДЕНИЕ 5
ГЛАВА 1. СИНТАКСИЧЕСКИЙ, СЕМАНТИЧЕСКИЙ И СЕМАНТИКО-СИНТАКСИЧЕСКИЙ АНАЛИЗ ТЕКСТОВ НА ЕСТЕСТВЕННОМ ЯЗЫКЕ 15
1.1. Синтаксический анализ 15
1.1.1. Задачи синтаксического анализа текстов на естественном языке. Модели синтаксической структуры предложения 15
1.1.2. Методы построения синтаксических деревьев составляющих 20
1.1.3. Методы построения синтаксических деревьев зависимостей 26
1.1.4. Системы синтаксического анализа текстов на русском языке 32
1.2. Семантический анализ 33
1.2.1. Модели семантики текста 34
1.2.2. Методы определения ролевых структур высказываний 45
1.3. Семантико-синтаксический анализ 53
1.3.1. Подходы, интегрирующие синтаксический и семантический анализ . 54
1.3.2. Подходы, интегрирующие построение синтаксических деревьев зависимостей с определением ролевых структур высказываний 56
1.4. Выводы 60
1.5. Цель и задачи исследования 62
ГЛАВА 2. ИНТЕГРАЦИЯ МЕТОДОВ СИНТАКСИЧЕСКОГО И СЕМАНТИЧЕСКГО АНАЛИЗА ТЕКСТОВ 64
2.1. Задача определения ролевых структур высказываний в текстах на русском языке 64
2.2. Метод определения ролевых структур высказываний в текстах на русском языке 67
2.2.1. Общий алгоритм работы семантического анализатора 67
2.2.2. Поиск предикатных слов 71
2.2.3. Поиск семантических аргументов 73
2.2.4. Назначение ролей семантическим аргументам 75

2.2.5. Разрешение неоднозначности назначения семантических ролей аргументам 81
2.3. Метод семантико-синтаксического анализа текстов 84
2.3.1. Общее описание метода семантико-синтаксического анализа 86
2.3.2. Поиск возможных семантических аргументов 90
2.3.3. Определение дополнительных семантических аргументов 92
2.3.4. Корректировка синтаксического дерева зависимостей 94
2.4. Выводы 102
ГЛАВА 3. ЭКСПЕРИМЕНТАЛЬНОЕ ИССЛЕДОВАНИЕ МЕТОДОВ СЕМАНТИЧЕСКОГО И СЕМАНТИКО-СИНТАКСИЧЕСКОГО АНАЛИЗА . 103
3.1. Тестовые данные и оценки качества 103
3.2. Оценка качества синтаксического анализа, определения категориально-семантических классов, морфологического анализа 105
3.3. Экспериментальное исследование метода поиска семантических аргументов 107
3.4. Экспериментальное исследования метода семантического анализа 109
3.5. Экспериментальное исследование метода семантико-синтаксического анализа 114
3.5.1. Оценка влияния метода семантико-синтаксического анализа на качество синтаксического анализа 114
3.5.2. Оценка влияния метода семантико-синтаксического анализа на качество решения задачи определения ролевых структур высказываний .. 123
3.6. Выводы 125
ГЛАВА 4. ПРИМЕНЕНИЕ МЕТОДОВ СЕМАНТИЧЕСКОГО И СЕМАНТИКО-СИНТАКСИЧЕСКОГО АНАЛИЗА К ЗАДАЧАМ ОБРАБОТКИ ТЕКСТОВ НА ЕСТЕСТВЕННОМ ЯЗЫКЕ 127
4.1. Применение семантического и семантико-синтаксического анализа для построения семантической сети реляционно-ситуационной модели 127
4.2. Применение семантического и семантико-синтаксического анализа для вопросно-ответного поиска 130
4.2.1. Метод ранжирования сниппетов для вопросно-ответного поиска в метапоисковой системе 133

4.2.2. Экспериментальное исследование метода ранжирования сниппетов для вопросно-ответного поиска в метапоисковой системе 144
4.3. Применение семантического и семантико-синтаксического анализа для решения задачи автоматического извлечения определений и авторских терминов из текстов научных публикаций 151
4.3.1. Метод извлечения определений и авторских терминов из текстов научных публикаций 151
4.3.2. Экспериментальное исследование метода извлечения определений и авторских терминов из текстов научных публикаций 156
4.4. Выводы 159
ЗАКЛЮЧЕНИЕ 161
СПИСОК СОКРАЩЕНИЙ И УСЛОВНЫХ ОБОЗНАЧЕНИЙ 162
СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ 163
ПРИЛОЖЕНИЯ 183
Приложение 1. Особенности реализации программного обеспечения для семантического и семантико-синтаксического анализа. Программная архитектура лингвистического процессора 183
Приложение 2. Исследование эффективности синтаксического анализатора на основе MaltParser 192
Приложение 3. Описание семантического словаря 198
Приложение 4. Примеры, в которых разработанная интегрированная система семантико-синтаксического анализа откорректировала синтаксическую и семантическую структуру предложения 204

ВВЕДЕНИЕ

Актуальность темы исследования.

Компьютерный лингвистический анализ текстов на естественном языке – перспективная быстроразвивающаяся область искусственного интеллекта. Одна из ключевых задач компьютерного лингвистического анализа заключается в построении такого структурированного представления текста, к которому можно применять методы и алгоритмы решения прикладных задач. Многие исследователи отмечают, что качественное решение таких задач, как вопросно-ответный поиск, извлечение информации и знаний из текстов, автоматическое реферирование, невозможно без проведения синтаксического и семантического анализа.

Существует значительное число разновидностей методов как синтаксического, так и семантического анализа, которые основаны на разных моделях синтаксической структуры предложения и различном понимании семантики. В настоящей диссертации исследуются методы построения синтаксических деревьев зависимостей и методы определения ролевых структур высказываний.

Деревья зависимостей моделируют синтаксическую структуру предложений в виде иерархии слов, связанных дугами, обозначающими синтаксическое подчинение между главным и зависимым словами. Подчинение обуславливается набором общих принципов, которые в целом сводятся к тому, что зависимое слово в предложении является уточняющим, необязательным, менее важным для передачи смысла высказывания, чем главное.

Модель семантики, основанная на ролевой структуре предложения, позволяет абстрагироваться от синтаксических деревьев предложений и сопоставлять разным грамматическим конструкциям одинаковые смысловые структуры. Задача определения ролевых структур высказываний (semantic role

labeling) включает в себя поиск предикатных слов, которые описывают в предложении некоторую ситуацию (это, например, глаголы, причастия, отглагольные существительные), поиск их семантических аргументов – синтаксических конструкций, которые выражают в предложении участников ситуации, а также определение значений аргументов, т.е. назначение им семантических ролей, которые играют участники в ситуации.

Задачи синтаксического и семантического анализа, как правило, решаются раздельно: сначала строится синтаксическая структура предложения, на основе которой затем строится семантическая структура. Для этого существует ряд методов, основанных как на правилах, так и на машинном обучении. Хотя современные методы позволяют добиваться достаточно хорошего качества решения этих задач, остается значительное пространство для улучшения этих методов. Анализ ошибок синтаксических и семантических анализаторов показывает, что, с одной стороны, для построения правильной синтаксической структуры предложения необходимы знания о его семантике, а, с другой стороны, ошибки в синтаксическом дереве негативно отражаются на качестве семантического анализа. Исходя из этого, некоторые исследователи выдвигают гипотезу о том, что совмещение синтаксического и сематического видов анализа может повысить их качество. Такой совмещённый подход к решению задачи глубокого лингвистического анализа будем называть семантико-синтаксическим анализом.

В настоящей диссертационной работе представлен новый метод семантико-синтаксического анализа, в котором интегрированы методы построения синтаксических деревьев зависимостей и определения ролевых структур высказываний. За счет информации, полученной на этапе семантического анализа предложения, корректируется синтаксическое дерево, что в свою очередь помогает исправить ошибки в ролевой структуре высказывания. Метод позволяет значительно повысить качество как синтаксического, так и семантического

анализа, что подтверждается проведенными экспериментами на размеченных русскоязычных корпусах текстов, а также улучшением качества решения прикладных задач обработки текстов.

Несмотря на то, что работы по созданию подходов, интегрирующих методы построения синтаксических деревьев зависимостей и методы определения ролевых структур высказываний, ведутся довольно давно, ранее не было предложено эффективного подхода, который позволил бы повысить как качество синтаксического, так и качество семантического анализа. Поэтому исследования в области семантико-синтаксического анализа, проведенные в настоящей диссертационной работе, являются актуальными.

Предмет исследования – методы и алгоритмы определения ролевых структур высказываний, а также методы и алгоритмы семантико-синтаксического анализа.

Целью исследования является повышение качества автоматического анализа текстов на естественном языке на основе интеграции методов синтаксического и семантического анализа.

Задачи исследования:

1. Провести исследование методов синтаксического и семантического анализа текстов на естественном языке.
2. Разработать метод определения ролевых структур высказываний в текстах на русском языке.
3. Разработать эффективный метод семантико-синтаксического анализа, в котором интегрированы методы построения синтаксических деревьев зависимостей и определения ролевых структур высказываний.
4. Реализовать методы семантического и семантико-синтаксического анализа. Разработать интегрированную систему семантико-синтаксического анализа.

5. Провести экспериментальные исследования методов семантического и семантико-синтаксического анализа.
6. Разработать методы решения прикладных задач, в которых используются результаты семантического и семантико-синтаксического анализа.
7. Провести экспериментальное исследование методов решения прикладных задач. Оценить влияние разработанных методов семантического и семантико-синтаксического анализа на качество решения этих задач.

Для решения поставленных задач применены следующие **методы исследования**:

1. Методы оптимизации.
2. Методы машинного обучения.
3. Методы компьютерной лингвистики.
4. Методы оценки качества алгоритмов машинного обучения.
5. Методы проверки статистической значимости полученных результатов.
6. Методы исследования качества синтаксического и семантического анализа.
7. Методы объектно-ориентированного проектирования программного обеспечения.

Научная новизна и результаты, выносимые на защиту.

1. Разработан новый метод автоматического определения ролевых структур высказываний, основанный на коммуникативной грамматике русского языка.
2. Разработан новый метод компьютерного семантико-синтаксического анализа текстов, в котором интегрированы методы построения синтаксических деревьев зависимостей и определения ролевых

структур высказываний, позволяющий повысить точность и полноту синтаксического и семантического анализа по сравнению с реализацией, в которой эти виды анализа выполняются раздельно.

3. Разработана и реализована интегрированная система семантико-синтаксического анализа. Система применена для решения задач вопросно-ответного поиска, извлечения определений и авторских терминов из текстов научных публикаций.
4. Экспериментально показано, что при использовании интегрированной системы семантико-синтаксического анализа существенно повышается точность вопросно-ответного поиска по сравнению с раздельным применением методов синтаксического и семантического анализа.
5. Экспериментально показано, что использование ролевой структуры предложения повышает эффективность построения правил для извлечения определений и авторских терминов из текстов научных публикаций.

Теоретическая значимость работы состоит в создании и экспериментальном исследовании новых методов интеграции и взаимодействия синтаксического и семантического видов анализа текстов на естественном языке.

Практическая значимость.

Разработанные методы семантического и семантико-синтаксического анализа являются основой для извлечения информации и знаний из текстов, вопросно-ответного поиска, автоматического реферирования и для решения других прикладных задач обработки текстов на естественном языке и информационного поиска.

Разработанное программное обеспечение, включающее реализацию методов семантического и семантико-синтаксического анализа текстов на естественном языке, **внедрено** в следующих системах:

1. информационно-поисковые сервисы портала «Руконт», «ООО Национальный цифровой ресурс «Руконт»»;
2. электронно-библиотечная система «Znanium.com», «ООО Научно-издательский центр ИНФРА-М»;
3. информационно-аналитическая система «Exactus Expert», «ЗАО РосИнтернет технологии»;
4. метапоисковая машина «Exactus», «Федеральное государственное учреждение науки ИСА РАН».

Результаты диссертационного исследования использованы при выполнении научно-исследовательских работ по следующим проектам Минобрнауки РФ, программам ОНИТ РАН и грантам РФФИ:

1. «Создание программного комплекса информационно-аналитической поддержки научно-технической деятельности на основе вычислительного семантического поиска и анализа неструктурированной текстовой информации» (в рамках ФЦП «Исследования и разработки по приоритетным направлениям развития научно-технологического комплекса России на 2007—2013 годы», ГК № 07.551.11.4003, 2011–2013 гг.).
2. «Исследование и разработка программного обеспечения понимания неструктурированной текстовой информации на русском и английском языках на базе создания методов компьютерного полного лингвистического анализа» (в рамках ФЦП «Исследования и разработки по приоритетным направлениям развития научно-технологического комплекса России на 2007—2013 годы», ГК № 07.514.11.4134, 2012–2013 гг.).
3. «Развитие методов и технологии семантического поиска и анализа научных публикаций Exactus Expert» (в рамках проекта 2.9 ОНИТ РАН 2012–2013 гг.).

4. «Исследование и разработка новых методов автоматического семантико-синтаксического анализа текстов, основанных на коммуникативной грамматике, реляционно-ситуационной модели текста и теории неоднородных семантических сетей» (в рамках проекта 12-07-33068 мол_а_вед РФФИ 2012–2013 гг.).
5. «Исследование и разработка методов извлечения целевой информации из первичных научных публикаций на основе реляционно-ситуационного анализа текстов и активного машинного обучения с использованием индуктивных и статистических моделей» (в рамках проекта 14-29-05023 офи_м РФФИ 2014 – 2016 гг.).

Достоверность результатов подтверждена экспериментальными исследованиями разработанных методов и алгоритмов.

Апробация результатов исследования.

Основные положения диссертации докладывались и обсуждались на следующих конференциях и семинарах:

1. XIII национальная конференция по искусственному интеллекту с международным участием (КИИ: Россия, Белгород, октябрь 2012 г.);
2. Workshop on Integrating IR technologies for Professional Search, in conjunction with the 35th European Conference on Information Retrieval (ECIR'13) (Россия, Москва, март 2013 г.).
3. Международная конференция «Диалог 2014», (Россия, Бекасово, июнь 2014 г.).
4. Шестая международная конференция «Системный анализ и информационные технологии», (Россия, Калининградская обл., г. Светлогорск, июнь 2015 г.).

Публикации.

Всего по теме исследования опубликовано 7 работ: 4 из них – в рецензируемых изданиях из списка ВАК РФ и приравненных к ним [1 – 4], 2

публикации в трудах международных и российских конференций [5, 6], 1 зарегистрированная программа для ЭВМ [7]. Опубликованные в этих работах результаты, относящиеся к методам семантического и семантико-синтаксического анализа, к методам вопросно-ответного поиска и методам извлечения определений и авторских терминов из текстов научных публикаций, получены лично автором.

Структура и объем работы.

Диссертация состоит из введения четырех глав, заключения, списка сокращений и условных обозначений, списка использованной литературы, а также четырех приложений. Полный объем диссертации составляет 210 страниц с 38 рисунками, 11 таблицами и 4 приложениями. Список литературы содержит 178 наименований.

В первой главе представлен аналитический обзор моделей синтаксических структур предложений и моделей семантики текстов на естественном языке, проанализированы проблемы автоматического синтаксического и семантического анализа, рассмотрены современные методы синтаксического и семантического анализа, а также подходы к интеграции этих методов в системах семантико-синтаксического анализа. Основное внимание уделено методам построения синтаксических деревьев зависимостей и определения ролевых структур высказываний. Детально проанализированы работы, в которых предлагаются способы интеграции методов построения синтаксических деревьев зависимостей и определения ролевых структур высказываний. Делаются выводы, обосновывающие актуальность исследования, формулируется цель и задачи исследования.

Во **второй главе** детализируется задача определения ролевых структур высказываний, подробно рассматриваются разработанные методы семантического и семантико-синтаксического анализа. Описываются методы поиска предикатных слов, семантических аргументов, назначения семантических

ролей аргументам с помощью семантического словаря, а также метод оптимального распределения семантических ролей между аргументами предикатных слов. Рассматриваются случаи, в которых информация о ролевой структуре предикатных слов может помочь при проведении синтаксического анализа. Описывается метод семантико-синтаксического анализа: этапы поиска потенциальных ошибок в синтаксической структуре с помощью метода определения ролевых структур высказываний, способы корректировки синтаксических деревьев, ограничения, накладываемые на добавление новых связей в синтаксическое дерево и способы их соблюдения, методы проверки исправлений с помощью классификаторов на основе машинного обучения и статистико-эвристического критерия, основанного на совместной встречаемости признаков слов.

В **третьей главе** описаны экспериментальные исследования разработанных методов семантического и семантико-синтаксического анализа текстов на естественном языке. Описываются тестовые и обучающие данные, методики оценки. Приводятся результаты экспериментальных исследований, проводится анализ ошибок, делаются выводы об эффективности разработанных методов.

В **четвертой главе** описываются подходы к решению прикладных задач обработки текстов на естественном языке, в которых используются результаты семантического и семантико-синтаксического анализа. Рассматриваются задачи построения семантической сети предложения, вопросно-ответного поиска, извлечения определений и авторских терминов из текстов научных публикаций. Проводятся экспериментальные исследования предложенных методов, оценивается влияние разработанных методов семантического и семантико-синтаксического анализа на качество решения прикладных задач.

В **заключении** приводятся основные результаты и выводы диссертационной работы.

В **приложении 1** описываются особенности реализации программного обеспечения для семантического и семантико-синтаксического анализа текстов на естественном языке. Приводятся его характеристики и примененные технологии. Рассматриваются требования, которым должен удовлетворять лингвистический процессор; описывается, реализованная в работе программная архитектура, позволяющая выполнить эти требования.

В **приложении 2** представлено описание семантического словаря. В **приложении 3** описаны эксперименты с обучаемым синтаксическим анализатором. В **приложении 4** приведены примеры, в которых разработанная интегрированная система семантико-синтаксического анализа откорректировала синтаксическую и семантическую структуру предложения. Сравниваются результаты, полученные с помощью интегрированной системы семантико-синтаксического анализа с результатами работы системы, в которой синтаксический и семантический анализ выполняются раздельно.

ГЛАВА 1. СИНТАКСИЧЕСКИЙ, СЕМАНТИЧЕСКИЙ И СЕМАНТИКО-СИНТАКСИЧЕСКИЙ АНАЛИЗ ТЕКСТОВ НА ЕСТЕСТВЕННОМ ЯЗЫКЕ

1.1. Синтаксический анализ

Синтаксический анализ занимает одно из важнейших мест в цепочке обработки текстов на естественном языке (ЕЯ) во многих программных приложениях. Он применяется в информационно-поисковых и информационно-аналитических системах, в системах машинного перевода и извлечения информации из текстов. На сегодняшний день синтаксический анализ является актуальной и активно исследуемой проблемой компьютерной лингвистики.

В этом разделе обсуждаются задачи синтаксического анализа текстов на ЕЯ, рассматриваются модели представления синтаксической структуры предложения, а также методы синтаксического анализа.

1.1.1. Задачи синтаксического анализа текстов на естественном языке. Модели синтаксической структуры предложения

В области обработки текстов на естественном языке, как и в информатике в целом, под синтаксическим анализом понимается сопоставление лексем некоторого языка (естественного или формального) с его формальной грамматикой. Единицей синтаксического анализа текстов на ЕЯ обычно является предложение.

С одной стороны, синтаксический анализ – это решение задачи распознавания. Он определяет, является ли предложение грамматически верным с точки зрения общепринятых правил построения фраз в некотором ЕЯ. Например, в английском языке артикль ставится перед определяемым словом, в русском языке определяемое и определитель должны быть согласованы в роде,

числе и падеже. Это востребовано в системах автоматической проверки качества текстов в текстовых редакторах.

С другой стороны, что более важно, в задаче понимания текста машиной синтаксический анализ – это построение такой структуры, которая позволяет приблизиться к некоторому эксплицитному формализованному представлению смысла текста. Но, в отличие от «глубокой» семантической структуры, которая строится в результате семантического анализа, синтаксическая структура обычно не связывает ЕЯ-конструкции с их значениями в некоторой предметной области. Синтаксическая структура может выступать либо как промежуточный результат, который является входом для семантического анализа, либо как удобное представление текста на ЕЯ для решения высокоуровневых прикладных задач, например, в информационно-аналитических системах или системах машинного перевода [8, 9].

Большинство моделей синтаксической структуры предложения ЕЯ опираются либо на грамматику составляющих, предложенной в работах Ноама Хомского [10], либо на грамматику зависимостей, для которой основополагающими считаются работы Люсьéна Теньéра [11] и Игоря Мельчука [12].

На начальном этапе исследований в области компьютерной лингвистики большее внимание уделялось грамматике составляющих. Эта модель предполагает, что предложение ЕЯ может быть представлено в виде иерархии составляющих – проективных синтаксических групп, которые не могут частично пересекаться, но которые в свою очередь состоят из более мелких групп (быть вложенными), вплоть до атомарных групп – слов предложения. Такую иерархическую структуру называют деревом составляющих (рисунок 1).

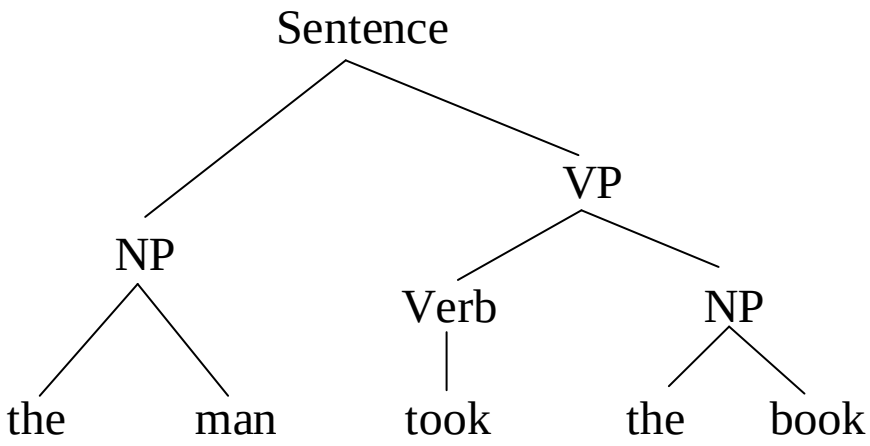


Рисунок 1 – Первое дерево составляющих, из работы Н. Хомского [10]

Идея о том, что слова в предложении ЕЯ группируются в составляющие, основывается на лингвистическом наблюдении того, что цепочки слов в предложении могут функционировать как единое целое и подчиняются единым грамматическим правилам. Составляющие можно перенести в середину или в конец предложения целиком, но частично их перенести без потери смысла нельзя.

Грамматика составляющих в иерархии Хомского – это контекстно свободная (КС) грамматика. Исследователи в целом согласны с тем, что естественные языки не являются регулярными, например, в работах [10, 13, 14] показывается, что английский язык не может быть распознан регулярными грамматиками. Исследователи также пришли к выводу, что существуют языки, содержащие конструкции, которые могут быть распознаны только с помощью контекстно зависимых грамматик. Например, в работах [15, 16] показывается, что в швейцарском немецком существуют конструкции, которые не распознаются КС-грамматиками. Тем не менее, хотя естественные языки могут выходить за рамки класса контекстно-свободных языков, моделирование синтаксических правил с помощью КС-грамматик дает хорошее приближение к реальности и позволяет решать большинство прикладных задач.

Грамматика зависимостей предполагает, что предложения текста можно структурировать в виде деревьев зависимостей, в которых слова связаны ориентированными дугами, обозначающими синтаксическое подчинение между главным и зависимым словом [17, 12, 18] (рисунок 2).

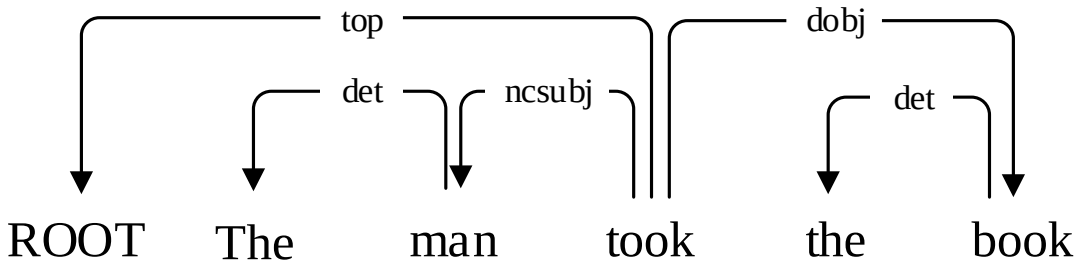


Рисунок 2 – Синтаксическое дерево зависимостей предложения «The man took the book» в формате ресурса «Universal Dependencies»[1] [19]

Главное отличие синтаксических деревьев зависимостей от деревьев составляющих в том, что здесь отсутствуют нетерминальные вершины, обозначающие составляющие, а синтаксические связи имеют пометки, которые обозначает их тип. Типы связей определяют грамматические функции слов в предложении или общие семантические отношения между словами.

Существующие грамматические теории зависимостей могут расходиться в том, по каким правилам устанавливаются связи, какие слова в этих связях являются управляющими, а какие зависимыми, а также какие существуют типы синтаксических связей. Можно привести множество традиционно спорных случаев. Например, в одних теориях считается, что предлог управляет словом, к которому он относится, в других теориях делается обратное предположение. Тем не менее, существует ряд общих принципов, которые обосновывают наличие синтаксической зависимости между словами в выражении на ЕЯ. В [20] Йоаким Нивре приводит некоторые из них:

[1] http://universaldependencies.github.io/docs/u/overview/syntax.html

- Управляющее слово определяет синтаксическую категорию выражения и часто может его заменить.
- Управляющее слово определяет семантическую категорию выражения, зависимое уточняет семантическую характеристику.
- Управляющее слово – обязательное, подчиненное – может быть необязательным.
- Управляющее слово выбирает зависимое и определяет, является ли оно обязательным.
- Форма зависимого слова согласуется с формой управляющего.
- Позиция зависимого слова зависит от управляющего.

Выбор деревьев зависимостей или деревьев составляющих для описания синтаксической структуры предложений ЕЯ связывают с двумя свойствами языков: проективностью их синтаксических конструкций и наличием свободного порядка слов.

Свойство проективности в терминах синтаксического дерева зависимостей означает: если все стрелки зависимостей проведены по одну сторону от прямой, на которой записано предложение, то ни одна из стрелок не пересекает никакую другую стрелку, а также никакая стрелка не накрывает корневой узел [21, 22]. В терминах деревьев составляющих проективность означает, что составляющие не могут разрываться на несколько отдельных, несмежных частей.

Грамматики составляющих широко применяется для анализа языков с фиксированным порядком слов, в которых мало конструкций, нарушающих свойство проективности, поскольку в грамматиках составляющих, как правило, не предусмотрены непроективные синтаксические отношения между словами. Например, такими языками являются английский, немецкий, турецкий.

Считается, что грамматики зависимостей хорошо отражают специфику языков со свободным порядком слов, в которых между словами может присутствовать значительное количество непроективных связей. К таким

языкам, относятся немецкий, чешский, русский, а также другие восточнославянские языки.

Отметим, что наличие свободного порядка слов в ЕЯ также влияет на трудоемкость его описания с помощью разных формализмов. Для анализа конструкций в языках со свободным порядком слов в грамматиках составляющих может потребоваться большое количество правил отдельно для каждого возможного случая расстановки слов [23], когда как правила грамматики зависимостей зачастую абстрагируются от порядка слов.

1.1.2. Методы построения синтаксических деревьев составляющих

В этом подразделе описываются проблемы, с которыми приходится сталкиваться при синтаксическом анализе текстов на естественном языке, из-за которых классические алгоритмы анализа по КС-грамматикам оказываются неприменимыми и кратко описываются эффективные алгоритмы, разработанные для решения этой задачи.

Методы построения синтаксических деревьев составляющих

Было показано, что, поскольку естественные языки с достаточной степенью общности можно приблизить контекстно-свободными языками, для их синтаксического анализа могут применяться контекстно-свободные грамматики и соответствующие алгоритмы для их распознавания. Однако от формальных КС-языков, таких как языки программирования, которые обычно могут быть эффективно проанализированы за линейное время от длины входной строки, естественные языки отличаются высокой неоднозначностью [23]. Неоднозначность может возникать на всех уровнях обработки текстов на ЕЯ. На этапе морфологического анализа может возникать несколько вариантов разбора слов, которые обладают различными морфологическими характеристиками (лексическая, частеречная, падежная омонимия, и др.). КС-грамматики ЕЯ обычно содержат большое число недетерминированных правил, что при разборе

с помощью классических алгоритмов приводит к большому числу возвратов и, как следствие, к чрезвычайно низкой вычислительной эффективности анализа. Помимо этого, грамматики ЕЯ обычно допускают несколько возможных вариантов разбора, т.е. допускают синтаксическую (или структурную) омонимию. Часто этот тип омонимии не может быть разрешен только лишь за счет лингвистических соображений и для определения подходящего разбора требует знаний о семантике слов и конструкций. Заметим, что нередки случаи, когда несколько вариантов синтаксического разбора правильны и с точки зрения семантики. Например, в предложении *«Мы встречали поэта из Грузии»* оба разбора, представленные на рисунке 3, допустимы с точки зрения семантики.

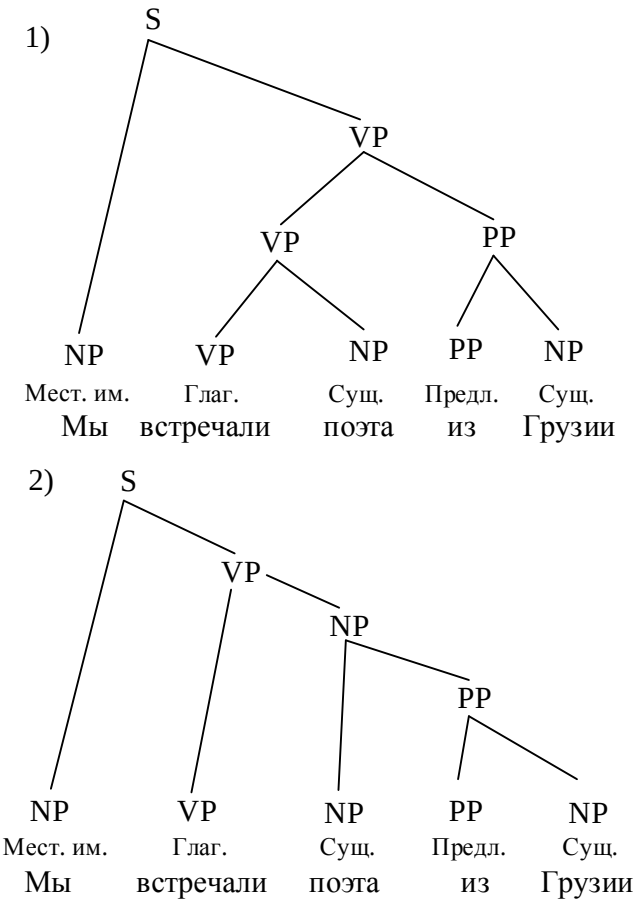


Рисунок 3 – Пример двух вариантов синтаксического разбора предложения «Мы встречали поэта из Грузии», допустимых с точки зрения семантики

Чтобы разрешить омонимию в подобных случаях требуются знания о прагматике текста или об его экстралингвистическом контексте [24]. В отсутствие знаний о семантике и прагматике текста синтаксический анализатор должен либо выдавать все возможные варианты разбора, либо стремиться найти лучший вариант в соответствии с некоторым критерием, опирающимся на уже построенные морфологические или синтаксические структуры предложения. Таким образом, неоднозначность ЕЯ делает классические алгоритмы разбора по КС-грамматикам неприменимыми.

Чтобы справиться с двумя проблемами, возникающими при анализе текстов на ЕЯ с помощью КС-грамматик: большим количеством возвратов и необходимостью сохранять несколько вариантов разбора, исследователи предложили ряд алгоритмов, использующих принципы динамического программирования. Основная идея применения динамического программирования при анализе с помощью КС-грамматик заключается в компактном хранении всех построенных в ходе разбора поддеревьев, чтобы при возникновении возврата не было необходимости заново воссоздавать когда-то уже построенные поддеревья.

Один из наиболее простых алгоритмов синтаксического анализа, использующих принципы динамического программирования – это алгоритм Кокка – Янгера – Касами (Cocke – Younger – Kasami, сокр. CYK, также встречается название CKY) [25, 26]. Для его работы необходимо преобразовать грамматику в нормальную форму Хомского, в которой все правила в правой части имеют либо терминал, либо пару нетерминалов. Эта форма позволяет при разборе использовать верхнетреугольную таблицу для хранения всех распознанных нетерминалов (синтаксических групп). Каждая ячейка *[i, j]* обозначает составляющую, которая включает в себя токены с *i*-ой позиции по *j*-ую, в ней хранятся все распознанные нетерминалы этой составляющей. Распознавание нетерминалов в ячейке *[i, j]* осуществляется путем сопоставления

нетерминалов из ячеек левее и ячеек ниже заданной с правилами грамматики. Алгоритм проводит разбор снизу-вверх, заполняя сначала левые и нижние ячейки. Предложение считается распознанным, если в правую верхнюю ячейку удалось занести нетерминал *S*, обозначающий предложение. Недостатком оригинального алгоритма CYK является то, что он работает только с КС-грамматиками, приведенными к нормальной форме Хомского. Поэтому, чтобы получить дерево составляющих в терминах исходной грамматики необходимо обратное преобразование, на что требуются дополнительные вычислительные затраты и затраты памяти.

Еще один алгоритм, использующий принципы динамического программирования, был предложен Джеем Эрли [27]. Этот алгоритм выполняет разбор сверху-вниз. Для хранения промежуточного дерева алгоритм использует таблицу состояний. Под состоянием понимается «правило с точкой», т.е. правило, в правой части которого отмечено на сколько оно выполнено в заданной позиции в тексте. Алгоритм последовательно просматривает предложение и заполняет таблицу состояний слева направо. Разбор начинается с добавления в левую ячейку таблицы правила, в левой части которого находится нетерминал *S*. Затем алгоритм в зависимости от текущих состояний и текущего слова выполняет действия: предсказание, сканирования, выполнение. Разбор считается выполненным, если в последней записи таблицы присутствует правило, в левой части которого находится нетерминал *S*.

Мартин Кей обобщил два вышеупомянутых подхода, представив схему построения эффективных алгоритмов разбора по КС-грамматикам – «chart parser» [28]. В этой схеме он предложил хранить частично построенные деревья в виде графа, а также ввел понятие стратегии. Изменяя стратегию, можно построить алгоритм разбора снизу-вверх типа CYK или сверху-вниз типа Эрли, или же реализовать более сложную стратегию, которая, например, на основе ряда эвристик могла бы выбирать следующее действие, меняя направление разбора.

Алгоритмы CYK, Эрли и другие разновидности «chart parser» в худшем случае обладают сложностью $O(n^3)$, где n – количество слов в предложении.

Принципиально другой подход для разбора по неоднозначным грамматикам был предложен в обобщенном восходящим магазинном анализаторе (Generalized Left-to-right Rightmost, сокр. GLR) Масару Томита [29]. Анализатор работает по схожим принципам, что и LR (left rightmost) анализаторы типа перенос-свертка, которые предназначены для разбора по детерминированным грамматикам. Но, чтобы эффективно обрабатывать неоднозначности в грамматике, в анализаторе GLR предусмотрена возможность эффективного разветвления стека. Как и в других рассмотренных анализаторах наличие разветвления в соответствии с принципами динамического программирования не приводит к дублированию информации на стеке за счет его представления в виде графа. Хотя в худшем случае сложность алгоритма GLR такая же, как и у разновидностей «chart parser» – $O(n^3)$, сложность распознавания у GLR зависит от степени недетерминированности грамматики. В случае, если грамматика детерминированная, сложность алгоритма – $O(n)$. Это выгодно отличает его от других рассмотренных алгоритмов.

Создание КС-грамматик для разбора ЕЯ языков является очень трудоемкой задачей. Для применения в реальных прикладных задачах грамматики должны содержать тысячи правил. Поддерживать и отлаживать эти грамматики весьма сложно. Еще одним принципиальным недостатком разбора с помощью вручную составленных КС-грамматик является тот факт, что несколько разборов могут быть верными с грамматической точки зрения, но недопустимы с точки зрения семантики. Чтобы выбирать правильные варианты разбора необходима возможность их ранжировать. С одной стороны, это позволяет найти лучший в соответствии с некоторым критерием результат, а с другой стороны, это может ускорить разбор, поскольку анализатор в случае неоднозначности будет направляться в наиболее весомые ветви, и, таким образом, скорее будет

построено наиболее адекватное синтаксическое дерево. Для ранжирования вариантов синтаксического разбора широкое применение нашли статистические подходы.

Наиболее известным статистическим подходом к задаче построения деревьев составляющих является применение стохастических контекстно-свободных (СКС) грамматик [30]. СКС-грамматики отличаются от обычных КС-грамматик тем, что каждому правилу грамматики назначена вероятность. Задачей синтаксического анализатора в этом случае является нахождение наиболее вероятного варианта разбора предложения. СКС-грамматики автоматически строятся на основе банков синтаксических деревьев – корпусов, в которых предложениям вручную или полуавтоматически сопоставлены синтаксические деревья. Например, для английского языка наиболее известным подобным банком является Penn Treebank [31]. Для разбора предложений на ЕЯ по СКС-грамматикам в основном применяют модификации, ранее представленных в этом разделе алгоритмов, использующих принципы динамического программирования (CYK, Эрли, «chart-parser»). Подходы, предложенные исследователями, различаются в основном методами моделирования вероятностей синтаксических деревьев, а также способами решения задачи поиска дерева с максимальной вероятностью.

Первые СКС-грамматики в основном опирались на синтаксические правила взаимодействия частей речи. Однако эксперименты показали, что статистической информации о сочетаемости частей речи недостаточно для разрешения неоднозначности ЕЯ. Оценки качества подобных анализаторов были значительно ниже, чем у анализаторов, использующих большие наборы эвристик. Развитием статистического подхода стали лексикализованные стохастические контекстно свободные грамматики [32]. В правилах этих грамматик помимо частей речи учитывается также лексика и статистика

встречаемости слов. Эта модификация значительно повысила качество статистических методов синтаксического анализа.

1.1.3. Методы построения синтаксических деревьев зависимостей

Первые методы построения деревьев зависимостей, как и методы построения деревьев составляющих были основаны на вручную созданных грамматиках, записанных в некотором формализме. Можно выделить несколько классов формализмов задания грамматик зависимостей и подходов к построению деревьев зависимостей.

К первому классу относятся формализмы, которые близки к формальным грамматикам Хомского [33, 34]. Они, как правило, способны моделировать лишь проективные языковые структуры. Для синтаксического анализа по таким грамматикам часто используют подходы и алгоритмы разбора, применяемые для построения деревьев составляющих – алгоритмы, использующие динамическое программирование для решения проблемы неоднозначности и недетерминированности грамматик, например, алгоритм Эрли или CYK [33, 35, 36].

Другой класс формализмов предполагает задание грамматик зависимостей в виде набора ограничений, что позволяет с помощью них моделировать как проективные, так и непроективные языковые структуры [18, 37]. Синтаксический анализ с точки зрения таких формализмов – это задача поиска в полном графе зависимостей дерева, которое удовлетворяет всем ограничениям грамматики. Заметим, что подобная задача является NP-полной, поэтому для ее решения применяют подходы, которые направленно ограничивают зону поиска правильных с точки зрения грамматики синтаксических деревьев.

Еще один класс подходов к представлению грамматик зависимостей заключается в задании некоторой системы правил (набора булевских функций) и детерминированного алгоритма разбора текста. Правила и алгоритм однозначно

задают синтаксическое дерево зависимостей. Этот подход был развит в работах Михаэля Ковингтона [38]. Он предложил простой алгоритм разбора, со сложностью $O(n^2)$: начиная с первого слова в предложении, сканировать слова слева направо и пытаться установить зависимость между текущим словом и всеми предыдущими словами (причем текущее слово может быть как главным, так и зависимым). М. Ковингтон показал, что этот алгоритм и набор булевских функций $f(w_1, w_2) = \begin{cases} true, \text{если между } w_1 \text{ и } w_2 \text{ можно установить связь} \\ false, \text{если между } w_1 \text{ и } w_2 \text{ нельзя установить связь} \end{cases}$ позволяют строить корректные синтаксические деревья, которые задаются другими более сложными формализмами грамматик зависимостей.

На сегодняшний день акцент исследований сместился с поиска наилучшей лингвистической теории и попыток ее формализации в виде грамматики зависимостей к поиску методов и подходов, так или иначе, использующих методы машинного обучения, и к моделированию с их помощью особенностей ЕЯ, присутствующих в обучающих данных. Применение методов машинного обучения с учителем для построения деревьев зависимостей показало ряд преимуществ по сравнению с подходами, основанными на вручную составленных грамматиках, среди которых – многократное снижение трудоемкости создания синтаксических анализаторов при наличие крупного размеченного корпуса. На сегодняшний день существует два основных подхода к построению деревьев зависимостей, в которых используются методы машинного обучения: графовый подход (graph-based dependency parsing) и синтаксический разбор на основе системы переходов (transition-based parsing).

Графовый подход к построению синтаксических деревьев зависимостей

Графовый подход предполагает, что строится модель, которая позволяет количественно оценивать синтаксические деревья предложения (графы зависимостей). Как правило, модель факторизует граф зависимостей на элементарные компоненты (в моделях первого порядка – это одинарные дуги, в

моделях второго порядка – это пара смежных дуг), которым назначаются веса. Задача синтаксического анализа в этом случае сводится к поиску в полном графе предложения дерева зависимостей с максимальной оценкой.

В первых работах, реализующих графовый подход, использовались порождающие модели для оценки графа зависимостей: проводилось моделирование совместной вероятности появления слов, дуг, и синтаксических меток дуг, а оценка всего дерева представлялась как произведение этих вероятностей [39]. В более поздних работах применяются дискриминантные модели оценки. Чаще всего используется аддитивная модель. В этом случае в модели первого порядка оценка всего графа складывается из весов его элементарных компонент и выражается следующим образом [21]:

$$score_tree\ (\text{y}) = \sum_{(i,j)\in y} score(i,j). \qquad (1)$$

Здесь y – граф зависимостей; (i,j) – ориентированная дуга графа y: $(i,j) \in y$, обозначающая синтаксическую зависимость.

Обычно вес дуги задается линейной функцией:

$$score(i,j) = \vec{w} \cdot \vec{f}(i,j). \qquad (2)$$

Здесь $\vec{w}$– вектор весов признаков – параметр модели; $\vec{f}(i,j)$ – функция получения вектора признаков дуги (i,j).

Пусть имеется предложение s и пусть *Y(s)* – множество всех деревьев, которые можно построить на предложении s. Задачу синтаксического анализа в введенных терминах можно записать следующим образом:

$$parse(s) = \underset{y\in Y(s)}{\operatorname{argmax}}\, score_tree\ (y, \vec{w}). \qquad (3)$$

Одной из первых работ, в которой описывается графовый подход к синтаксическому анализу, является работа Джейсона Айзенера [39]. В ней он предложил порождающую модель оценки графа зависимостей и алгоритм синтаксического анализа, схожий с алгоритмом CYK. Он позволяет строить (или находить в полном графе предложения) проективные синтаксические деревья зависимостей за время $O(n^3)$.

Позже в работе [40] было показано, что дискриминантные модели могут успешно применяться вместе с алгоритмом Айзенера для построения проективных деревьев. В [21] предложена модификация, которая в рамках графового подхода позволила строить непроективные деревья. Вместо алгоритма Айзенера был использован алгоритм поиска максимального остовного дерева в графе [41, 42]. Этот анализатор известен в литературе как MST parser (maximum spanning tree parser). Было обнаружено, что сложность построения произвольного (непроективного) синтаксического дерева с помощью алгоритма поиска максимального остового дерева составляет $O(n^2)$, что меньше чем сложность алгоритма Айзенера, строящего проективные деревья. Исследователи также показали на примере чешского языка, что эта модификация позволяет добиваться лучшего качества синтаксического анализа для языков со свободным порядком слов, которые допускают большое количество непроективных связей. При этом скорость анализа выше по сравнению с методом, в котором применяется алгоритм Айзенера. Однако применение алгоритма поиска максимального остовного дерева в MST анализаторе может приводить к незначительным уменьшениям качества анализа для языков, в которых количество непроективных связей мало (например, в английском).

В последующих работах, проведенных в рамках графового подхода к синтаксическому анализу, исследовалось применение моделей оценки графа зависимостей второго порядка [43 – 46]. Было показано, что поиск

непроективных деревьев зависимостей в случае использования модели второго порядка является NP-полной задачей [47]. Поэтому исследования в этом направлении в основном были ограничены поиском методов построения только проективных деревьев. В работе [43] предложен алгоритм синтаксического анализа за время $O(n^3)$, а в работе [45] – за $O(n^4)$ при использовании модели второго порядка.

Метод синтаксического анализа на основе системы переходов

В методе синтаксического анализа на основе системы переходов строится и обучается модель для оценки переходов от одного состояния синтаксического анализатора к другому состоянию. В общем случае алгоритм анализа сводится к предсказанию на основе построенной модели действия анализатора и перехода из текущего состояния в новое состояние. Предсказания и переходы осуществляются до тех пор, пока не будет построено синтаксическое дерево зависимостей.

Этот подход был изначально предложен в работах [48, 49]. В них использовался алгоритм типа перенос-свертка со сложностью $O(n^2)$, позволявший строить проективные синтаксические деревья зависимостей. Для предсказания переходов анализатора использовался метод опорных векторов. Отметим, что в качестве признаков использовались части речи и леммы связываемых слов и их контекста.

Подход на основе системы переходов был развит в работах Йоакима Нивре [50, 51]. Он предложил алгоритм для построения проективных синтаксических деревьев за время $O(n)$. В отличие от работ [48, 49] анализатор Й. Нивре позволяет не только строить синтаксические деревья предложений, но также определять тип связей и назначать им метки. Это позволило включить в модель для предсказания переходов признаки, которые можно извлечь из частично построенного синтаксического дерева. Изначально в анализаторе применялись метрические методы классификации для предсказания переходов, однако в

дальнейших исследованиях их место занял метод опорных векторов, поскольку было показано, что он дает лучшие результаты.

Алгоритм Нивре последовательно обрабатывает слова предложения слева направо. Слово, его контекст и частично построенное синтаксическое дерево задают признаки, которые классификатор на основе машинного обучения использует для предсказания следующего действия (перехода) анализатора. В алгоритме используется стек и предусмотрены следующие типы переходов:

- Left-Arc – установить зависимость между входным словом (главное) и словом на вершине стека (подчиненное), вытолкнуть одно слово из стека;
- Right-Arc – установить зависимость между словом на вершине стека (главное) и входным словом (подчиненное), положить входное слово в стек;
- Reduce – вытолкнуть одно слово из стека (допускается только в том случае, если слово на вершине стека имеет родителя);
- Shift – положить входное слово в стек (допускается только в том случае, если входное слово, т.е. анализатор не пришел в конец предложения).

При выполнении действия Right-Arc и Left-Arc анализатор обращается к классификатору также для определения типа синтаксической связи. Алгоритм гарантирует построение проективного дерева. Если допустить, что в случае, когда анализатору необходимо выполнить действие Reduce, и слово на вершине стека не имеет родителя, анализатор все равно выталкивает слово из стека и делает его подчиненным вспомогательной вершине ROOT, то построенные деревья также будут полностью связанными.

Обучение классификатора, предсказывающего переходы, проходит аналогично синтаксическому анализу. Анализатор просматривает слова предложения слева направо и определяет действие, которое необходимо

предпринять на текущем шаге, чтобы построить синтаксическое дерево из обучающего корпуса. Это действие и набор признаков используются как обучающий пример.

В работе [52] представлена также модификация анализатора, которая позволяет снять ограничение на проективность синтаксического дерева. Суть этой модификации состоит в том, что все непроективные связи преобразуются в проективные связи со специальными метками. Затем уже полностью проективное дерево используется для обучения. В ходе анализа метки расшифровываются и построенное анализатором проективное дерево преобразуется в непроективное.

Подход к синтаксическому анализу на основе системы переходов был реализован в программном пакете MaltParser [53, 54]. Этот пакет включает в себя различные алгоритмы синтаксического анализа, включая алгоритм Нивре и алгоритм Ковингтона, а также реализации нескольких методов машинного обучения для предсказателей переходов. Из-за своей эффективности и производительности MaltParser на сегодняшний день является одним из наиболее широко используемых синтаксических анализаторов. Исследования по применению MaltParser проводились для многих языков, в том числе и для русского [55, 56, 3].

1.1.4. Системы синтаксического анализа текстов на русском языке

Большой вклад в развитие синтаксических анализаторов на русском языке внесли работы по созданию систем машинного перевода. Поскольку для того, чтобы добиться приемлемого качества перевода, необходим синтаксический анализ, разработчикам подобных системы приходилось вкладывали большие силы в создание синтаксических анализаторов текстов на русском языке. Одна из наиболее известных систем в этой области была создана коллективом под руководством Ю.Д. Апресяна: ЭТАП-1, ЭТАП-2, ЭТАП-3 [57, 8]. Несмотря на долгую историю создания этих систем, проект ЭТАП-3 продолжает развиваться,

в том числе в сторону применения статистических методов [58]. В системах машинного перевода Retrans и МетаФраз, разработанных коллективном под руководством Г.Г. Белоногова, реализован семантико-синтаксический анализатор текстов на русском языке [59 – 61]. Стоит отметить поверхностный синтаксический анализатор программного пакета AOT.ru[2], который разрабатывался для системы машинного перевода Dialing [62].

Работы в области синтаксического анализа текстов на русском языке в настоящий момент продолжаются коллективом под руководством Т.Ю. Кобзаревой, Д.Г. Лахути [63, 64], а также многими другими исследователями, например, [65, 66].

Важной вехой в области синтаксического анализа текстов на русском языке стало создание Синтаксически размеченного корпуса русского языка СинТагРус [67]. Это позволило успешно применить к задаче синтаксического анализа подходы на основе машинного обучения, которые показали весьма высокую эффективность [58, 55]. По-видимому, в будущем статистические методы будут играть все большую роль в синтаксическом анализе текстов на русском языке.

1.2. Семантический анализ

Компьютерный семантический анализ тесно связан с задачей понимания текста машиной. Существует много интерпретаций понятия «смысл текста» и задачи его понимания. Например, по Д.А. Поспелову [68] система понимает введенный в нее текст, если с точки зрения человека (или группы экспертов) она правильно отвечает на вопросы, связанные с информацией, заключенной в тексте. Здесь речь может идти не о простом получении фактов, которые явно присутствуют в тексте, а о выявлении скрытых смыслов, которые вносит автор. Д.А. Поспелов выделяет несколько уровней понимания текста, с точки зрения

[2] http://aot.ru/

сложности вопросов, на которые должна иметь возможность отвечать интеллектуальная система. Руководствуясь определением из работы [68], можно рассматривать смысл текста как описание знаний, содержащихся в нем, на формальном языке представления знаний, который позволяет решать достаточно широкий круг задач, связанных с анализом текста, а задачу семантического анализа – как трансляцию ЕЯ-выражений на язык представления знаний. В качестве языка представления знаний текста на ЕЯ могут выступать, например, язык предикатов первого порядка, семантические сети, фреймы. Заметим, что формальное представление смысла текста, получаемое в результате семантического анализа, отличается от поверхностных синтаксических или морфологических структур тем, что оно явно представляет знания об объектах и связях между ними. Эти объекты и связи часто выходят за рамки лингвистических категорий и могут отражать либо объекты и связи в реальном мире, либо концепты и отношения между ними в некоторой предметной области. Для компьютерного семантического анализа помимо знаний о грамматике ЕЯ, как правило, требуется множество других типов знаний, включая экстралингвистические (значения слов, значения, ассоциированные с грамматическими конструкциями, знания о структуре дискурса, знания о теме дискурса, знания о мире), а также методы рассуждения на них [69].

В этом разделе рассматривается ряд моделей семантики текста, а также современные методы семантического анализа. Акцент делается на задаче определения ролевых структур высказываний и методах ее решения.

1.2.1. Модели семантики текста

В области обработки текстов на естественном языке существует множество моделей семантики текста. Несмотря на более полувековую историю исследований в этой области проблема моделирования семантики текста на ЕЯ до сих пор является открытой.

Формальная семантика

Одним из наиболее известных способов моделирования семантики выражений на естественном языке является их представление на языке логики. Ричард Монтегю первый показал, что английский язык можно транслировать в формальный язык [70]. Его предположение заключалось в том, что на теоретическом уровне формальные и естественные языки существенным образом не различаются, а значение любого предложения есть условие его истинности. При этом значение предложения складывается (выводится) из значений его частей. Монтегю разработал грамматику семантики, в основу которой легло исчисление предикатов первого порядка, расширенное интенсиональной логикой, которая включает в себя лямбда-исчисление, расширенную структуру типов и типы функций.

Формальная семантика – это один из подходов к семантике синтаксиса. В работе [71] была выдвинута гипотеза о том, что каждое синтаксическое правило имеет свой аналог в семантике, и формальную грамматику для синтаксиса и семантики можно совместить. На рисунке 4 представлен пример дерева, которое построено на основе применения формальной интегрированной синтаксической и семантической грамматики, в соответствии с которым осуществляется вывод формулы высказывания «Every restaurant closed».

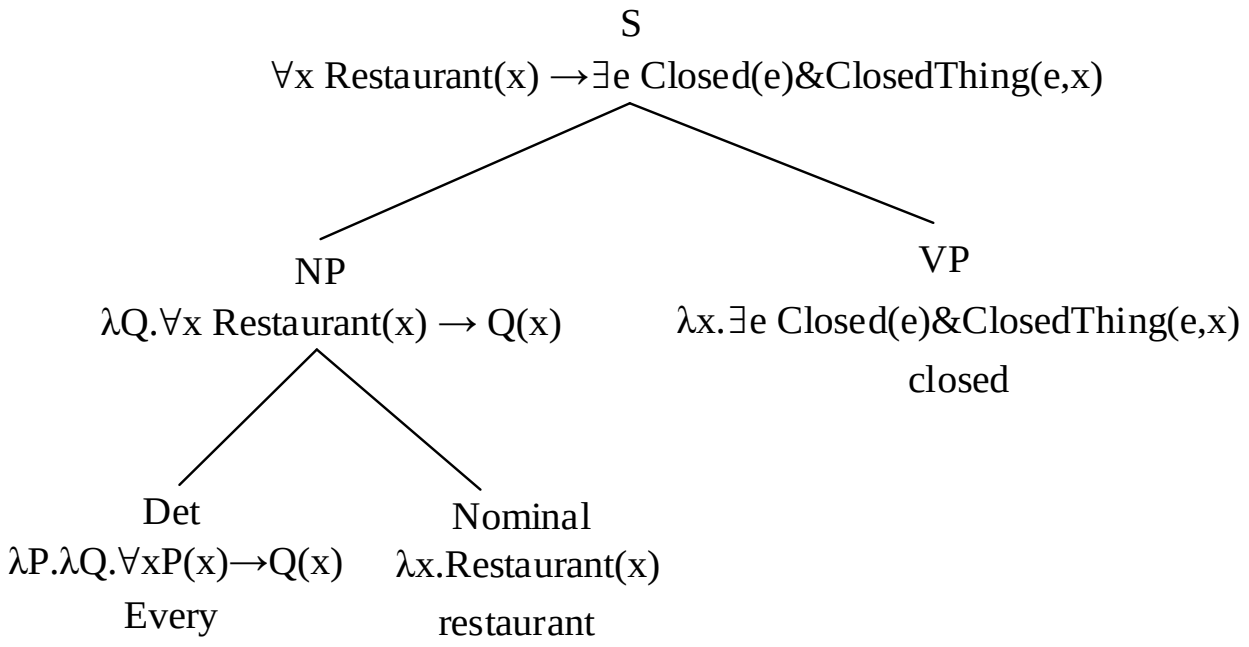


Рисунок 4 – Результат разбора по формальной синтаксической и семантической грамматике предложения «Every restaurant closed» [69]

Работа Р. Монтегю дала начало целому направлению, в рамках которого исследуются подходы моделирования на формальном языке временных аспектов ЕЯ, выражений событий, характеристик объектов в тексте, идиом, метафор и др. На практике известны попытки применения формальной семантики в задаче машинного перевода [72]. Однако на сегодняшний день, исследователи пришли к выводу, что из-за неоднозначности и гибкости ЕЯ применение аппарата формальной семантики на практике, с одной стороны, проблематично из-за его сложности, а с другой стороны, недостаточно для решения современных задач обработки текстов на ЕЯ.

Модель семантики текста, основанная на ролевой структуре предложения

Одной из самых старых моделей семантики текста на ЕЯ считается модель, основанная на ролевой структуре предложения. Современная формулировка этой модели была предложена в работах Чарльза Филлмора [73]. Он ввел понятие

глубинного падежа – универсальной (тематической) роли, которую могут исполнять участники ситуации, обозначенной в тексте глаголом. Глубинные падежи было предложено рассматривать как «универсальные врожденные понятия, идентифицирующие типы суждений, которые человек способен делать о событиях, происходящих вокруг него, – суждений о вещах такого рода, как «кто сделал нечто», «с кем нечто случилось», «что подверглось некоему изменению»» [74]. Всего было введено около двенадцати подобных понятий, которые, как предполагалось, позволяют представить в обобщенном виде все возможные смыслы участников ситуаций. На сегодняшний день среди исследователей нет согласия о составе инвентаря ролей и о том, по каким принципам их необходимо выделять и различать. Тем не менее, существует ряд ролей, которые довольно часто используются в литературе (таблица 1) [22].

Таблица 1 – Некоторые часто используемые тематические роли

Тематическая роль	**Определение**
Агенс	Одушевленный инициатор действия, контролирующий его.
Пациенс	Участник, больше остальных вовлеченный в действие и претерпевающий в ходе него наиболее существенные изменения.
Бенефактив	Участник ситуации, чьи интересы каким-то образом затронуты в процессе ее осуществления, получающий от нее пользу или вред.
Экспериенцер	Участник – носитель чувств и восприятий, переживающий действие.
Стимул	Источник информации при глаголах чувственного восприятия или источник непроизвольного переживания.
Инструмент	Неодушевленный объект, с помощью которого осуществляется действие, но который сам не претерпевает изменений при его осуществлении.
Адресат	Получатель сообщения.
Источник	Место, из которого осуществляется движение.
Цель	Место, в которое осуществляется движение.

Ч. Филлмор сделал важное наблюдение, которое состоит в том, что одни и те же роли (глубинные падежи) могут быть по-разному выражены в поверхностных грамматических структурах ЕЯ – синтаксических деревьях и морфологических характеристиках (грамматических падежах). Таким образом, модель семантики текста, основанная на ролевой структуре предложения, позволяет абстрагироваться от синтаксической и морфологической структуры предложений и разным грамматическим конструкциям ставить в соответствие одинаковые роли (рисунок 5).

Рисунок 5 – Иллюстрация того, что одинаковые роли могут соответствовать разным грамматическим конструкциям

Модель семантики текста, основанная на ролевой структуре предложения, была развита во многих работах как отечественных, так и зарубежных лингвистов: [75 – 78] и др. Предложения текста в этой модели можно представить в виде совокупности предикатных слов, их аргументов и семантических ролей [22, 79].

- Под предикатным словом (ПС) подразумевается лексема или синтаксическая конструкция, обладающая набором семантических валентностей. Такими словами являются глаголы, отглагольные существительные, причастия и некоторые другие части речи. ПС в тексте обозначают ситуацию.
- Семантическими аргументами называются языковые выражения в тексте, которыми заполняются семантические валентности предикатных слов. Семантические аргументы обозначают в тексте участников ситуаций, обозначаемых ПС.
- Семантическая роль – часть семантики предиката, отражающая общие свойства аргумента предиката. Это значение синтаксической конструкции (аргумента) при предикатном слове.
- Предикатное слово и его аргументы формируют предикатно-аргументную (ПА) структуру. Между предикатным словом и его семантическим аргументом установлена предикатно-аргументная связь.
- Одним из важнейших свойств модели семантики текста, основанной на ролевой структуре предложения, является то, что обычно (за исключением случаев с симметричными глаголами) у одного предикатного слова не может быть двух аргументов с одной и той же семантической ролью, и аргументы могут иметь только одну роль при заданном предикатном слове.

Модель семантики текста, основанная на ролевой структуре предложения, развивалась как в сторону обобщенных ролей [80], так и в сторону более специфичных ролей, определенных для конкретных предикатных слов или групп предикатных слов – фреймов. В современных системах обработки текстов для многих языков чаще всего применяется модели семантики ресурсов PropBank [81] и FrameNet [82].

Модель ресурса PropBank предполагает, что каждое значение предикатного слова имеет набор специфичных для него ролей. Роли в PropBank не имеют имени, а лишь нумеруются: Arg0, Arg1, …, Arg5. Ролям Arg0 и Arg1 обычно соответствуют обобщенные агенс и пациенс, смысл остальных ролей зависит от предикатных слов. Помимо нумерованных ролей в PropBank предусмотрены обобщенные роли Argm-*, которые являются универсальными для всех предикатных слов. Например, Argm-loc соответствует месту действия; Argm-dir – направлению действия; Argm-tmp – времени или продолжительности действия. Ресурс PropBank состоит из текстов банка синтаксических деревьев Penn Treebank, размеченных семантическими ролями глаголов, и словаря, в котором для глаголов английского языка указан список нумерованных ролей с комментариями и примерами. Ресурс NomBank предоставляет аналогичную разметку для предикатных слов-существительных [83]. Существует много ресурсов аналогичных оригинальному PropBank на других языках (например, на китайском [84], на арабском [85]).

В отличие от PropBank, ресурс FrameNet задает семантические роли не для отдельных глаголов, а для фреймов, которые объединяют в себя несколько близких по смыслу предикатных слов. В один фрейм могут попадать слова разных частей речи: глаголы, существительные, наречия и др. Роли во FrameNet разделяются на центральные (core roles) – специфичные для конкретного фрейма и периферийные (non-core roles) – могут возникать в разных ситуациях (время, место, продолжительность и др.). Еще одной важной особенностью FrameNet является то, что он фиксирует отношения между разными фреймами и семантическими ролями. Фреймы могут быть связаны отношениями наследования, причины-следствия, перспективизации (один фрейм рассматривает событие с более узкой точки зрения, чем другой нейтральный фрейм), часть-целое и др. FrameNet, как и PropBank предоставляет семантически размеченный корпус текстов на английском языке. Существуют ресурсы типа

FrameNet и для других языков, например, для немецкого [86], для испанского [87]. Ведутся работы по созданию схожего ресурса и для русского языка (FrameBank [78]).

В области обработки текстов на ЕЯ, на сегодняшний день задача определения ролевых структур высказываний (semantic role labeling) является одной из центральных задач в цепочке анализа текста. Она известна также как поверхностный семантический анализ. Эта задача претерпела несколько переформулировок. В работе Даниэля Журавски и Даниэля Галдиа [88] была поставлена задача назначения ролей синтаксическим группам (аргументам) при заданных в тексте предикатных словах. В дальнейшем формулировка была обобщена, на семинарах CoNLL Shared Task 2004 [89] и 2005 [90] аргумент мог не совпадать с синтаксической группой и для его нахождения необходимо было применять специальные методики. Наконец эта задача была сформулирована в терминах лексических зависимостей (CoNLL Shared Task 2008 [91] и 2009 [92]). Задача определения ролевой структуры высказываний включила в себя поиск предикатных слов, определение их значений и установление семантической зависимости между предикатным словом и словом-аргументом, где тип связи обозначает семантическую роль (рисунок 6).

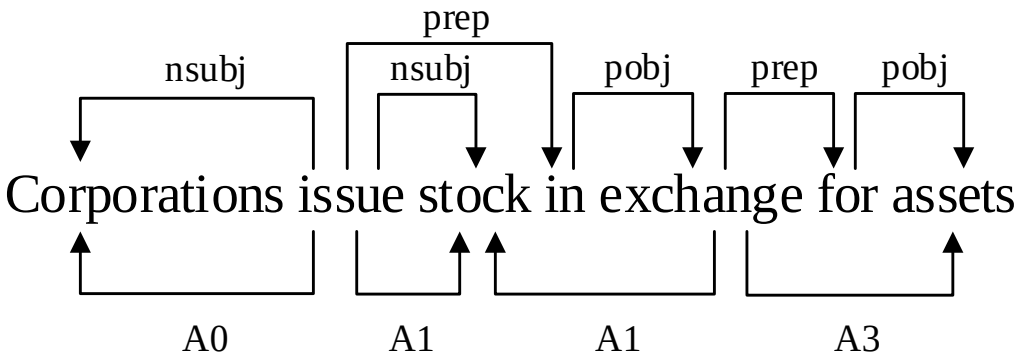


Рисунок 6 – Пример результата работы синтаксического анализатора (сверху) и результата работы анализатора, который определяет ролевую структуру в соответсвии с моделью PropBank (снизу)[3]

Модель семантики текста, основанная на ролевой структуре предложения, нашла применение в широком спектре систем обработки текстов на ЕЯ: системах вопросно-ответного поиска [93], извлечения информации [94, 95], автоматического реферирования [96], машинного перевода [97, 98] и др.

Реляционно-ситуационная модель

Реляционно-ситуационная модель текста формализует глубинные семантические структуры текста с помощью аппарата неоднородных семантических сетей [99]. Реляционно-ситуационная модель опирается на теорию коммуникативной грамматики Г.А. Золотовой [100].

В теории коммуникативной грамматики важную роль играет понятие синтаксемы. Синтаксема представляет собой минимальную синтактико-семантическую единицу языка, несущую обобщенный, категориальный смысл и характеризующуюся взаимодействием морфологических, семантических и функциональных признаков [101, 74]. Категориальный смысл синтаксемы называется ее значением. В отличие от ролей, которые в лингвистических теориях чаще всего ассоциируются с аргументами предикатных слов (например,

[3] http://www.cliffsnotes.com/more-subjects/accounting/accounting-principles-ii/corporations/accounting-for-stock-transactions

таких как глаголы) и тесно с ними связаны, значения синтаксем в этой теории являются в некоторой степени автономными. Это рассуждение вытекает из того, что многие конструкции в естественных языках имеют смысл даже в отрыве от контекста и предикатного слова. Например, «в лесу», «за лесом», «над лесом» могут обозначать место действия; «в лес» – конечную точку действия; «из леса» – начальную точку действия; «топором» – инструмент действия. Кроме того, в языке имеется множество выражений, в которых предикатные слова отсутствуют. Например, хотя в выражении «дом в лесу» отсутствуют глаголы или другие предикатные слова, ясно, что синтаксема «дом» имеет значение некоторого объекта, а синтаксема «в лесу» – это место нахождения.

Реляционно-ситуационная модель формализует значения синтаксем предложения текста на ЕЯ и их семантические отношения в виде семантической сети. Вершинами этой сети являются синтаксемы. Синтаксемы могут быть двух видов – именные синтаксемы (выраженные именной или предложной группой) и предикатные синтаксемы. Значение предикатных синтаксем – это смысл предиката, который они выражают. Значения именных синтаксем в тексте можно представить в виде: связей между именными синтаксемами и предикатными синтаксемами, связей между именными синтаксемами и служебной вершиной SemRoot, где тип связи определяет значение.

Связи со служебной вершиной сети позволяют формализовать те значения синтаксем, которые не связаны с предикатным словом, как в примере «дом в лесу» (рисунок 7).

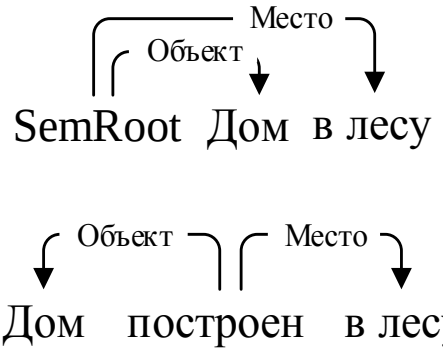


Рисунок 7 – Представления значений именных синтаксем в реляционно-ситуационной модели

Реляционно-ситуационная модель также формализует семантические отношений между синтаксемами. В семантической сети им соответствуют связи между вершинами, обозначающими именные синтаксемы. Семантические отношения между синтаксемами – это отношения между концептами в некоторой предметной области, которые можно выявить в тексте. В примере «*Ошибка врачей привела к эпидемии*» между синтаксемой «Ошибка» и «эпидемия» есть причинно-следственная связь, которую можно выявить в тексте. Семантическая сеть на рисунке 9 фиксирует это важное отношение, когда как простая модель, основанная на ролевой структуре предложения, представленная на рисунке 8 – нет.

Рисунок 8 – Представление семантики текста с помощью модели, основанной на ролевой структуре предложения

Рисунок 9 – Упроещнное представление семантики текста с помощью реляционно-ситуационной модели

Модель семантики в корпусе PropBank позволяет создавать такие представления смысла, которые абстрагируются от синтаксической структуры. Модель семантики во FrameNet позволяет дополнительно провести абстракцию от конкретных предикатных слов в тексте с помощью понятия фрейма. Реляционно-ситуационная модель семантики текста добавляет еще один уровень абстракции в виде отношений между любыми синтаксемами в предложении. Кроме того, она позволяет фиксировать значения синтаксем в конструкциях без предикатных слов. Таким образом, эта модель задает более глубокое представление семантики текста, чем в классических ролевых моделях.

Реляционно-ситуационная модель нашла применение для решения многих задач извлечения информации и информационного поиска [2, 102, 103].

1.2.2. Методы определения ролевых структур высказываний

В этом подразделе представлены два основных подхода к решению задачи определения ролевых структур высказываний: с помощью машинного обучения с учителем на семантически размеченных корпусах и подход, в котором применяются различные методы обучения без учителя (кластеризация) или с частичным привлечением учителя.

Методы на основе машинного обучения с учителем

Создание больших семантически размеченных корпусов для ряда языков позволило применить к решению задачи определения ролевых структур

высказываний методы машинного обучения с учителем. Основополагающей в этой области считается работа Даниэля Журавски и Даниэля Галдиа [88]. В ней на корпусе ресурса FrameNet обучался статистический анализатор для назначения семантических ролей синтаксическим составляющим. В дальнейшем методы определения ролевых структур на основе машинного обучения с учителем были развиты во многих работах. Применялись новые методы машинного обучения [104, 105], разрабатывались новые признаки для классификации [106], проводились эксперименты с различными схемами декомпозиции задачи на подзадачи [107], исследовались различные алгоритмические подходы [108].

В ходе исследований в области определения ролевых структур высказываний был сформирован общий подход к решению этой задачи. Чаще всего решение разбивают на несколько этапов:

- Поиск предикатных слов в предложении.
- Поиск семантических аргументов предикатных слов в предложении.
- Назначение ролей каждому аргументу независимо от других аргументов.
- Общая оценка и выбор наилучшей семантической структуры для всех аргументов заданного предиката или всего предложения.

Поиск предикатных слов в предложении обычно реализуется с использованием словарей лемм или по морфологическим признакам (например, предикатными словами считаются все глаголы). В целом эта подзадача считается простой, мало исследуется, и для ее решения применяются простые методы. На этом этапе иногда применяются бинарные классификаторы для определения того, является ли глагол или существительное предикатным словом, а также набор n-арных классификаторов для разрешения семантической омонимии предикатных слов для заданной леммы [109, 110].

Этап определения семантических аргументов может начинаться с отсеивания слов-кандидатов с помощью набора простых эвристик. К примеру, эвристики, предложенные в работе [106], позволяют значительно сократить количество кандидатов при поиске семантических аргументов в текстах на английском языке, что упрощает дальнейшую задачу определения аргументов с помощью классификатора на основе машинного обучения.

Первоначально задача выявления аргументов рассматривалась как проблема сегментации предложения в контексте синтаксического дерева составляющих. В этой формулировке аргумент – это сегмент, состоящий из нескольких слов, который нужно правильно выделить в предложении и связать с нужным предикатным словом (в общем случае сегмент не обязательно является непрерывным). В дальнейшем для унификации с задачей построения дерева зависимостей была предложена другая постановка, в которой необходимо определить только слово-вершину синтаксической группы, которая является семантическим аргументом, и связать эту вершину с предикатным словом. В этой постановке слово-вершина однозначно идентифицирует аргумент, поэтому под аргументами часто подразумевают только слова-вершины.

При решении задачи выявления аргументов в обоих формулировках широко применяют бинарные классификаторы. Для каждого отобранного слова предложения и для заданного предикатного слова классификатор предсказывает наличие между ними предикатно-аргументной связи. В качестве признаков для определения семантических аргументов в текстах на английском языке используют части речи предикатного слова и классифицируемого слова, позицию классифицируемого слова относительно предикатного слова, залог (пассивный / активный) предикатного слова, путь в синтаксическом дереве от классифицируемого слова к предикатному (типы синтаксических связей, либо составляющих), длина пути и другие контекстные признаки. Во многих работах

для определения семантических аргументов и для назначения им ролей используется одинаковый набор признаков, в этом случае он, как правило, шире.

Назначение семантической роли аргументу осуществляется с помощью n-арного классификатора, где n – количество ролей. Признаковое пространство на этом этапе может быть гораздо шире, чем при выявлении сематического аргумента: помимо признаков, использующихся на предыдущем этапе добавляются много новых. Признаки предикатного слова: лемма, класс в WordNet [111], класс предикатного слова (например, определенный с помощью кластеризации по лексике), синтаксический фрейм [88, 112]. Признаки аргумента: леммы из синтаксической группы семантического аргумента в виде мешка слов, n-граммы частей речи [113]. В работе [94] в качестве признака предлагается выбирать из группы семантического аргумента наиболее «информативное» слово и его признаки (лемма, часть речи, класс именованной сущности, к которой это слово относится). Наиболее информативное слово определяется с помощью эвристик и может не совпадать с синтаксической вершиной группы семантического аргумента. В [106] в качестве признака предлагается использовать место аргумента в синтаксической структуре предложения относительно других аргументов.

Разработка новых информативных признаков для задачи определения ролевых структур высказываний в текстах на различных языках является предметом глубоких исследований. Хотя некоторые признаки (например, лемма предикатного слова) используются почти повсеместно, отличительные особенности естественных языков, как правило, требуют создания специфичных признаков для каждого языка. Например, в языках с развитой морфологией, таких как славянские, дополнительно необходимо учитывать признаки, отвечающие за согласование слов, в то же время признаки, связанные с порядком слов, могут быть неинформативными. Определение семантических аргументов и назначение им ролей может быть объединено в единой модели для машинного обучения. В

этом случае обучается n-арный классификатор, в котором n – количество ролей плюс метка для обозначения отсутствия роли.

После всех предыдущих этапов для каждой роли каждого аргумента в отдельности определены оценки независимо от других ролей и аргументов. Эти «локальные» оценки используются на следующем этапе определения глобальной оценки всей семантической структуры, связанной с предикатным словом или со всем предложением.

На этапе глобальной оценки разрешается неоднозначность распределения ролей по семантическим аргументам, и семантическая структура приводится в соответствие с ограничениями модели семантики текста: каждый семантический аргумент может иметь только одну роль, и одна и та же роль не может быть назначена двум аргументам. При этом ищется такая структура, глобальная оценка которой является оптимальной или просто «хорошей» в соответствии с некоторым критерием. Для решения этих подзадач применяют методы линейного целочисленного программирования [114], а также модели для ранжирования ролевых структур, которые выбирают наилучший результат среди заданных кандидатов (например, модели на основе условных случайных полей [115]). Существуют также статистические модели для совместной оценки всей семантической структуры предложения или предикатного слова. В [116] предлагается модель на основе скрытой Марковской модели для совместной оценки выявления предикатных слов, их аргументов и назначения семантических ролей аргументам. В [117] используются модель на основе условных случайных полей для совместного выявления семантических аргументов и назначения им ролей. В работе [118] описывается модель для совместного выявления предикатных слов, аргументов и назначения им семантических ролей, а также для разрешения семантической омонимии предикатных слов.

Для повышения качества определения ролевых структур существует ряд подходов, в которых выходы нескольких семантических анализаторов

объединяются в один финальный результат. За счет того, что в разных анализаторах заложены разные модели для машинного обучения или они обучены на разных корпусах или используют разные параметры обучения, комбинация их выходов позволяет повысить устойчивость и полноту анализатора на данных, сильно отличающихся от обучающей выборки [119, 120].

Стоит отметить, что для большинства методов определения ролевых структур высказываний предварительно необходимо провести полный синтаксический анализ с построением либо синтаксического дерева составляющих, либо синтаксического дерева зависимостей. Было показано, что качество синтаксического анализа оказывает сильное влияние на качество определения ролевой структуры [114, 113]. Тем не менее, существуют работы, в которых предлагаются методы семантического анализа, для которых требуется предварительно провести лишь поверхностный синтаксический анализ предложения [121].

Методы машинного обучения с учителем показали эффективность в задачах установления семантических ролей для многих языков, для которых были созданы размеченные семантическими ролями корпуса, включая английский, испанский, китайский, немецкий, чешский и др. [92]. Много исследований проводится на англоязычном корпусе PropBank, а также на его аналогах на других языках.

Оценки качества для задачи определения ролевых структур высказываний сильно разнятся для разных языков. Приведем оценки F_1-меры из разных современных работ: для испанского: 68 – 77%; для чешского: 67 – 80%; для китайского: 70 – 76%; для английского: 80 – 83%. Чем больше ресурсов, таких как семантически размеченные корпуса и тезаурусы, разработано для естественного языка, тем выше удается исследователям поднять качество решения этой задачи.

Исследований, посвященных задаче автоматического определения ролевых структур в текстах на русском языке, довольно мало. Наиболее близкими к этой области являются работы создателей ресурса FrameBank [78, 122], система для семантической интерпретации текстов, связанных с экономической и общественно-политической деятельностью персон и организаций [123], а также работы автора диссертации [3, 4]. Среди других достаточно близких работ стоит отметить анализатор, для построения поверхностного семантического графа А. Сокирко [124] и исследования по установлению значений синтаксем [125]. Описание применения методов машинного обучения для определения ролевых структур высказываний в текстах на русском языке, а также оценка этих методов присутствуют только в работе [125] и работах автора диссертации [3, 4]. Малое количество исследований по применению методов машинного обучения к задачам определения ролевых структур высказываний в текстах на русском языке в значительной степени связано с недостатком размеченных корпусов, которые могли бы играть роль обучающей выборки и золотого стандарта при тестировании. Проект FrameBank призван решить эту проблему. Однако на текущий момент он находится в разработке, в нем нет размеченных полных текстов, которые необходимы для эффективного применения методов машинного обучения и их объективной оценки.

Методы на основе машинного обучения с частичным привлечением учителя и без учителя

Недостатком методов определения ролевых структур высказываний, использующих машинное обучение с учителем, является большая зависимость этих методов от наличия и качества семантически размеченных обучающих корпусов. Чтобы обеспечить приемлемое качество анализа, обучающие корпуса должны быть весьма объемными. Создание подобных корпусов очень трудоемкая и дорогостоящая работа. Поэтому для многих языков, в том числе и русского, обучающие корпуса, пригодные для разработки анализаторов,

выполняющих установление семантических ролей, отсутствуют. Стоит также отметить, что при анализе текстов из одной предметной области анализатором, обученным на корпусе из другой предметной области, качество анализа заметно снижается. Таким образом, чтобы эффективно применять методы машинного обучения с учителем при решении задачи определения ролевых структур высказываний, необходимо не только создавать корпусы для интересующего языка, но и адаптировать их для каждой интересующей предметной области.

Поэтому в последнее время исследователи ищут способы для снижения зависимости от вручную размеченных корпусов. Предлагаются подходы, основанные на машинном обучении с частичным привлечением учителя (semi-supervised machine learning) и на машинном обучении без учителя (unsupervised machine learning).

Ряд подходов с частичным привлечением учителя известен под названием «проекция аннотаций». В этих подходах семантическая структура размеченных предложений на одном языке сопоставляется схожим по некоторым признакам неразмеченным предложениям, переведенным на другой язык. Предпринимались попытки перевести английские семантически размеченные корпуса на немецкий [126], шведский [127], китайский [128], французский [129]. Для определения семантического фрейма предикатных слов, отсутствующих в словаре, зачастую применяют тезаурус WordNet [111]. В работе [130] проводится обобщение фреймов за счет наличия гипонимии между известными и неизвестными предикатами в тезаурусе. Ряд исследователей [131, 132] трактуют задачу определения фрейма для неизвестного предикатного слова как задачу классификации, в которой для расчета близости предикатных слов также используется WordNet. В работе [133] на основе небольшого семантически размеченного корпуса строится большой размеченный корпус, который используется для обучения семантических анализаторов. Новые аннотации строятся путем переноса предикатно-аргументных структур предложений из

размеченного корпуса на близкие к ним предложения из неразмеченного корпуса. Близость предложений вычисляется путем сопоставления их синтаксических деревьев и лексических признаков.

Подходы к решению задачи определения ролевых структур высказываний на основе машинного обучения без учителя используют кластеризацию [134 – 136]. В этом случае целью ставится разделить все аргументы на кластеры таким образом, чтобы каждый кластер соответствовал некоторой семантической роли. При этом, как правило, нет заранее заданного набора ролей или их признаков. Набор семантических ролей и характерные им свойства формируются автоматически в процессе кластеризации. Полученные кластеры и их признаки, с одной стороны – могут быть использованы напрямую для решения прикладных задач (например, для вопросно-ответного поиска), а с другой стороны, могут быть применены для быстрой семантической разметки корпуса, поскольку вручную экспертам достаточно разметить не весь текст, а лишь сопоставить роли самим кластерам.

1.3. Семантико-синтаксический анализ

Идея об интеграции компьютерного синтаксического и семантического анализа высказывалась многими исследователями. На сегодняшний день достаточно очевидно, что из-за многозначности конструкций в ЕЯ для построения синтаксических деревьев с высокой точностью нельзя ограничится лишь знаниями о грамматике языка в узком смысле – т.е. лингвистическими знаниями о морфологии слов и о синтаксических правилах разбора конструкций, которые оперируют только такими лингвистическими категориями как части речи и морфологические признаки. Чтобы хорошо решать эту задачу в общем случае необходимы статистические знания о ЕЯ, знания о семантике ЕЯ-конструкций, а в некоторых случаях о прагматике дискурса.

Структурная неоднозначность ЕЯ потребовала для синтаксического анализа разработки специальных анализаторов, статистических методов и подходов с применением методов машинного обучения. Во многих работах по синтаксическому анализу отмечается большая важность лексических признаков для достижения высоких показателей качества разбора. Поэтому, в современных синтаксических анализаторах текстов на ЕЯ часто учитываются не только сочетания частей речи, морфологических признаков и структурных признаков, но также лексическая сочетаемость слов друг с другом, что в некотором смысле является формой знаний о семантике языка [137].

Но существуют подходы, которые идут гораздо дальше простого учета лексем, лемм, коллокаций или семантических классов слов на этапе синтаксического анализа. Некоторые исследователи предлагают альтернативы традиционному подходу, когда синтаксический разбор, в котором строится синтаксическое дерево, строго предшествует семантическому анализу, в котором формируется смысловое представление текста, и разрабатывают подходы, совмещающие эти два вида анализа в одной процедуре. Анализ текста на ЕЯ, который выполняется в соответствии с этими подходами, будем называть семантико-синтаксическим[4].

1.3.1. Подходы, интегрирующие синтаксический и семантический анализ

Семантико-синтаксический анализ объединяет в себе подходы, в которых применяется как тесная интеграция синтаксического и семантического анализа, например, в едином алгоритме, так и подходы, в которых они разделены, но результаты, полученные на двух этапах анализа, влияют друг на друга.

Стремление к тесной интеграции основывается на предположении, что наличие доступа во время анализа к синтаксическим и семантическим знаниям о

[4] Заметим, что некоторые исследователи используют термин семантико-синтаксический анализ (или синтактико-семантический) в отношении совокупности синтаксического и семантического анализа в смысле конвейерного подхода.

ЕЯ, а также к обеим частично построенным структурам, позволит разрешать языковые неоднозначности раньше, чем в традиционном последовательном подходе. С одной стороны, за счет этого может достигаться преимущество в вычислительной эффективности обработки, поскольку в совместном анализе не нужно порождать лишние варианты синтаксического разбора и хранить их до разрешения неоднозначности при семантическом анализе. Но, с другой стороны, параллельный семантический анализ может существенным образом затормозить общий анализ. Поскольку многие некорректные варианты разбора могут отсеиваться только лишь за счет синтаксического анализа, для подобных вариантов выполнение параллельного семантического анализа превращается в дополнительные накладные расходы.

Ряд работ по семантико-синтаксическому анализу посвящено тесной интеграции грамматик составляющих с формальной семантикой. Такие грамматики называют синхронными. Правила этих грамматик расширяются семантическими выражениями на языке представления знаний. Таким языком может служить, например, расширенный язык предикатов первого порядка (с выражениями лямбда-исчисления). Определяются правила преобразования семантических выражений в соответствии с синтаксическим деревом. Семантико-синтаксический анализ может выполняться с помощью модифицированных алгоритмов построения деревьев составляющих. Например, в [69] показывается, что для этих целей может быть использован модернизированный алгоритм Эрли. В работах [138, 139] используются статистические синтаксические анализаторы для преобразования ЕЯ-выражений в запросы к базе данных. В работе [140] предлагается алгоритм типа CYK для совместного синтаксического и семантического анализа текста по стохастической грамматике «Head-Driven Phrase Structure Grammar» [141]. Существуют анализаторы, которые строят глубокие семантические структуры сразу, без промежуточных поверхностных синтаксических структур. Таким

образом, в них полностью стирается граница между синтаксическим и семантическим анализом.

Исследователи уделяют внимание проблеме интеграции синтаксического анализа и определения ролевых структур высказываний. Хотя в этом подходе интеграция между семантическим и синтаксическим чаще всего является менее тесной чем в вышеупомянутых методах, этот подход представляет наибольший интерес для настоящей работы. Он будет рассмотрен отдельно в разделе 1.3.2.

Рассмотрим работы по семантико-синтаксическому анализу текстов на русском языке. Подход, развиваемый коллективном Г.Г. Белоногова, предполагает тесную интеграцию синтаксического и семантического анализа [59]. Разработчики коммерческой технологии ABBYY Compreno [9] утверждают, что в их анализаторе синтаксическая и семантическая структуры строятся при анализе текста параллельно. В системе ЭТАП-3 [8] синтаксический анализатор при разборе использует семантические признаки слов и конструкций, однако семантическая структура строится на последующих этапах. Синтаксический анализатор, представленный в работе [65], основывается на синтактико-семантическом словаре Тузова [142]. В ходе синтаксического анализа используется семантические признаки слов для разрешения морфологической и структурной омонимии. В частности, используется информация о сочетаемости классов слов из словаря Тузова. Хотя этот анализатор авторы называют семантико-синтаксическим, непосредственно семантический анализ он не выполняет.

1.3.2. Подходы, интегрирующие построение синтаксических деревьев зависимостей с определением ролевых структур высказываний

Даниэль Журавски и Даниэль Галдиа [88] первыми предложили подход, в котором определение ролевых структур высказываний интегрировано с синтаксическим анализом. В этом подходе с помощью синтаксического

анализатора, основанного на СКС-грамматике, для каждого предложения строилось большое количество вариантов синтаксического разбора. Для каждого из вариантов выбиралось наилучшее распределение ролей, а затем выбирался вариант с наибольшей общей вероятностью. Эксперименты показали небольшой прирост полноты определения ролевых структур на 0,7% с 61,2% до 61,9%, при этом точность выявления аргументов незначительно уменьшилась на 0,2%.

Большой вклад в развитие методов семантико-синтаксического анализа внесли участники соревнований CoNLL Shared Task 2008 [91] и CoNLL Shared Task 2009 [92], в которых оценивалось качество и синтаксического (построение деревьев зависимостей), и семантического анализа (определение ролевых структур высказываний) совместно. В 2008 году задача ставилась только для одного языка – английского, в 2009 году участники должны были выполнить анализ для еще шести языков: каталанского, китайского, чешского, немецкого, японского и испанского. Шесть команд, участвовавших в соревновании в 2008 году, и четыре команды, участвовавших в соревновании в 2009 году, реализовали методы, которые так или иначе совмещают синтаксический и семантический анализ.

В работах [143, 144] описывается система, которая участвовала в соревновании в 2008 году, а в работе [145] представлена ее модификация, которая участвовала в соревновании в 2009 году. Семантико-синтаксический анализ в этих системах реализуется путем построения дерева зависимостей одновременно с установлением семантических меток синтаксическим связям. Сначала проводится предварительный синтаксический анализ, после чего проводится основной совмещенный анализ. Для предварительного и основного синтаксического анализа применялся графовый подход и алгоритм Айзенера [39]. В совмещенном анализе вес синтаксической связи с семантической пометкой складывался из веса синтаксической связи и нормированного веса семантических меток (ролей) от разных предикатных слов.

Строя синтаксическое дерево одновременно с назначением семантических меток, анализатор максимизирует совокупный вес семантико-синтаксической структуры. Экспериментальное тестирование этого метода показало, что совмещенный анализ не влияет на качество синтаксического анализа, но повышает качество определения ролевых структур (на 4,9% по F_1 мере) по сравнению с системой, в которой синтаксический и семантический анализ выполняются раздельно.

Средний результат на соревновании в 2008 году и один из лучших результатов на соревновании в 2009 году показала система, описанная в работах [146 – 148]. Авторы применили анализатор на основе системы переходов с использованием метода перенос-свертка для параллельного построения отдельной синтаксической и семантической структуры предложения. Для предсказания действий анализатора была обучена модель, которая представляет собой динамическую байесовскую сеть, последовательно изменяющую свою структуру на основе частично построенного вывода [149]. В качестве признаков задействовались уже построенные в ходе вывода синтаксические и семантические зависимости. Таким образом, модель при обучении максимизирует совместную вероятность синтаксических и семантических зависимостей. Эксперименты, проведенные в работах, показали, что при отсутствии взаимодействия между синтаксической и семантической структурой в процессе вывода существенно ухудшается качество семантического анализа (на 3,5% по F_1-мере). Авторы также показали, что интеграция семантического и синтаксического анализа незначительно снижает точность последнего (на 0,2%).

В [150] было применено смешивание результатов восьми синтаксических анализаторов, построенных на основе MaltParser с помощью варьирования алгоритмов и направлений разбора, в единое комбинированное синтаксическое дерево. Подобный шаг позволил значительно улучшить качество синтаксического анализа по сравнению с результатами лучшего единичного

анализатора. Были построены две различные системы для семантического анализа. Кроме того, на основе MaltParser была создана система семантико-синтаксического анализа, которая одновременно строит синтаксическое дерево зависимостей и назначает связям семантические метки. Синтаксическое дерево, построенное системой семантико-синтаксического анализа, смешивалось с комбинированным синтаксическим деревом. Новое комбинированное дерево направлялось на вход двум системам семантического анализа. Окончательная семантико-синтаксическая структура строилась путем смешивания результатов двух систем семантического анализа с результатом системы семантико-синтаксического анализа.

В [151] был реализован подход, в котором синтаксический и семантический анализ выполняются последовательно, а затем они повторяются, при этом на новой итерации используются признаки, полученные из уже построенной семантико-синтаксической структуры. В работе проводилось до двух итераций цепочек анализа. Эксперименты показали, что этот подход может заметно улучшить первоначальный результат. Однако улучшения наблюдались не для всех языков, и не всегда лучший результат достигался на последней итерации. Авторы заключают, что эффект от повторного анализа зависит от качества первоначального анализа. Семантико-синтаксический анализ в этой работе позволил повысить точность синтаксического анализа до 0,3%.

В [109] описывается система, которая на соревновании в 2008 году достигла лучших совместных оценок качества синтаксического и семантического анализа. В системе этапы анализа выполнялись последовательно. В результате строилось несколько вариантов синтаксических деревьев, для каждого из которых строились предикатно-аргументные структуры. Затем выполнялась выбирающая окончательный вариант процедура переранжирования результатов, обученная максимизировать совместную оценку синтаксического и семантического анализа. Эксперименты показали, что процедура переранжирования немного

улучшает результаты и синтаксического (точность на 0,14%), и семантического анализа (на 0,5% по F_1-мере), и, соответственно, совместные оценки. Схожий подход использовался также в работе [152].

Авторы [153] реализовали подход, в котором интегрированы этапы назначения меток синтаксическим связям и поиска семантических аргументов с использованием Марковской модели максимальной энтропии [154].

В [155] описывается система, в которой реализовано полностью совместное обучение установлению синтаксических и семантических зависимостей. Для этого использовались метрические классификаторы: один предсказывал наличие связи между токенами, другие два – наличие связи и ее метку (синтаксическую метку или семантическую роль). Результаты классификаторов комбинировались с помощью процедуры ранжирования.

Среди недавних публикаций по семантико-синтаксическому анализу помимо трудов семинаров CoNLL Shared Task следует отметить работу [156]. В ней предлагается подход для совместной максимизации оценки синтаксической и семантической структур. Это позволило повысить качество определения ролевых структур высказываний более чем на 2% по F_1-мере по сравнению с системой, в которой эти два вида анализа выполнялись раздельно.

1.4. Выводы

В качестве модели синтаксической структуры предложения для семантико-синтаксического анализа было выбрано синтаксическое дерево зависимостей. Выбор модели на основе зависимостей вместо составляющих во многом обусловлен тем, что для построения деревьев зависимостей существуют более эффективные методы. Например, алгоритм Нивре отрабатывает за линейное время от длины предложения $O(n)$, что значительно быстрее, чем алгоритмы анализа по КС-грамматикам, которые работают за $O(n^3)$ в общем случае. Кроме

того, деревья зависимостей более адекватно отражают синтаксическую структуру в языках с произвольным порядком слов, таких как русский. В современных исследованиях в области синтаксического анализа акцент сместился с создания сложных формализмов задания грамматик на разработку подходов, использующих машинное обучение. При разработке этих подходов строятся модели, отражающие особенности и структуру ЕЯ, которые можно восстановить по обучающим данным, а также создаются эффективные алгоритмы применения этих моделей. Наличие крупного синтаксически размеченного корпуса текстов на русском языке СинТагРус позволяет применить для синтаксического анализа текстов на русском языке обучаемые синтаксические анализаторы. В настоящей работе предлагается для синтаксического анализа использовать один из самых эффективных обучаемых анализаторов MaltParser.

В качестве теоретической основы для представления семантики текста в семантико-синтаксическом анализе была выбрана модель на основе ролевой структуры предложения. Эта модель, несмотря на свою простоту, нашла применение во многих прикладных задачах обработки текстов на естественном языке.

Обзор методов семантико-синтаксического анализа показывает, что ранее был предложен ряд подходов к интеграции методов построения деревьев зависимостей и методов определения ролевых структур высказываний для анализа текстов на английском языке. Многие исследователи сообщают о том, что системы, в которых реализовано взаимодействие между синтаксическим и семантическим анализом, позволяют качественнее решать задачу определения ролевых структур высказываний, чем аналоги, в которых эти этапы выполняются раздельно. Однако за счет подобной интеграции исследователям, как правило, не удается добиться улучшения качества синтаксического анализа. Во многих работах сообщается даже о том, что его точность немного снижается. Таким

образом, проблема построения системы, в которой информация о ролевой структуре предложения могла бы эффективно быть использована на этапе синтаксического анализа остается актуальной.

Работ, в которых решаются задачи, схожие с определением ролевых структур высказываний в текстах на русском языке, довольно мало, оценки методов представлены лишь в нескольких работах и работах автора диссертации [3, 4]. Результаты исследований по созданию системы семантико-синтаксического анализа для русского языка, в которой интегрируются методы построения синтаксических деревьев зависимостей на основе машинного обучения и методов определения ролевых структур высказываний опубликованы только в работах автора диссертации [3 – 5].

1.5. Цель и задачи исследования

Целью настоящего исследования является повышение качества синтаксического и семантического анализа текстов на естественном языке за счет разработки метода семантико-синтаксического анализа, в котором эти два вида анализа интегрированы. Исследование в первую очередь направлено на создание методов анализа текстов на русском языке.

Для достижения необходимо:

1. Провести исследование методов синтаксического и семантического анализа текстов на естественном языке.
2. Разработать метод определения ролевых структур высказываний в текстах на русском языке.
3. Разработать эффективный метод семантико-синтаксического анализа, в котором интегрированы методы построения синтаксических деревьев зависимостей и определения ролевых структур высказываний.

4. Реализовать методы семантического и семантико-синтаксического анализа. Разработать интегрированную систему семантико-синтаксического анализа.
5. Провести экспериментальные исследования методов семантического и семантико-синтаксического анализа.
6. Разработать методы решения прикладных задач, в которых используются результаты семантического и семантико-синтаксического анализа.
7. Провести экспериментальное исследование методов решения прикладных задач. Оценить влияние разработанных методов семантического и семантико-синтаксического анализа на качество решения этих задач.

ГЛАВА 2. ИНТЕГРАЦИЯ МЕТОДОВ СИНТАКСИЧЕСКОГО И СЕМАНТИЧЕСКГО АНАЛИЗА ТЕКСТОВ

2.1. Задача определения ролевых структур высказываний в текстах на русском языке

Задача определение ролевых структур высказываний, как и большинство других задач компьютерной лингвистики, имеет несколько интерпретаций. Нет единого мнения о том, какие конструкции естественного языка считать аргументами предикатных слов, какие роли должны назначаться аргументам: обобщенные, тематические, специфичные для предикатных слов (PropBank), специфичные для фреймов (FrameNet), и тем более нет единого мнения по поводу инвентаря семантических ролей. Поэтому, чтобы сформулировать постановку задачи определения ролевых структур высказываний, необходимо конкретизировать все вышеупомянутые аспекты, а также определить формальную структуру, которая в результате должна быть построена анализатором.

Постановка задачи и методы ее решения в настоящей диссертации в значительной степени опираются на семантический словарь, разработанный в Институте системного анализа РАН экспертами в области теоретической лингвистики [157]. Основной целью его создания являлись сбор и формализация знаний о семантике предикатных слов и синтаксем русского и английского языков в виде, в котором они могли бы быть использованы в компьютерных системах обработки текстов на ЕЯ. В семантическом словаре хранятся словарные статьи, каждая из которых соответствует некоторой ситуации. В словарной статье находятся леммы предикатных слов (глаголов и девербативов), обозначающих одну и ту же ситуацию. Словарная статья также содержит семантические значения, которые могут иметь синтаксемы при заданных

предикатных словах, а также признаки, которыми должны обладать синтаксемы, чтобы им были сопоставлены те или иные значения. В качестве основных признаков используются падеж, предлог, а также категориально-семантический класс существительного. Под категориально-семантическим классом (КСК) существительных подразумевается их обобщенное значение. Подробное описание структуры семантического словаря представлено в приложении 3. Некоторые схожие ресурсы для русского языка описаны в [78, 158, 159].

Хотя словарь создавался в терминах теории коммуникативной грамматики Г.А. Золотовой [100], он может быть использован и в задаче определения ролевых структур высказываний. Значения синтаксем в этом словаре можно рассматривать как широкий набор семантических (тематических) ролей. Словарные статьи семантического словаря в некоторой степени схожи с фреймами ресурса FrameNet – они задают список предикатных слов и их центральные роли. В постановке задачи определения ролевых структур высказываний в настоящей диссертационной работе будем полагать, что список семантических ролей, предикатных слов и их ролевые структуры заданы семантическим словарем.

Будем использовать понятие семантического аргумента в терминах работы [91]. В ней задача определения ролевых структур высказываний формально ставится как установление маркированной зависимости между предикатным словом и вершиной синтаксической группы, являющейся аргументом предикатного слова. Таким образом, задача определения аргумента сводится к определению слова-вершины синтаксической конструкции, значение которой определяется, поэтому под аргументом можно понимать отдельное слово с привязкой к некоторой позиции в синтаксическом дереве предложения. Уточним также, что в качестве семантических аргументов мы будем рассматривать только существительные, местоимения, вопросительные слова и числительные.

Задачу определения ролевых структур высказываний будем понимать в наиболее широком смысле, т.е. будем в нее включать выявление предикатных слов, выявление их аргументов, а также назначение аргументам семантических ролей.

На вход семантическому анализатору подается дерево синтаксических зависимостей предложения, морфологические характеристики слов, а также категориально-семантические классы слов (на рисунке 10 представлен пример синтаксического дерева из корпуса СинТагРус [67]). На выходе должен быть построен граф семантических зависимостей предложения (рисунок 11).

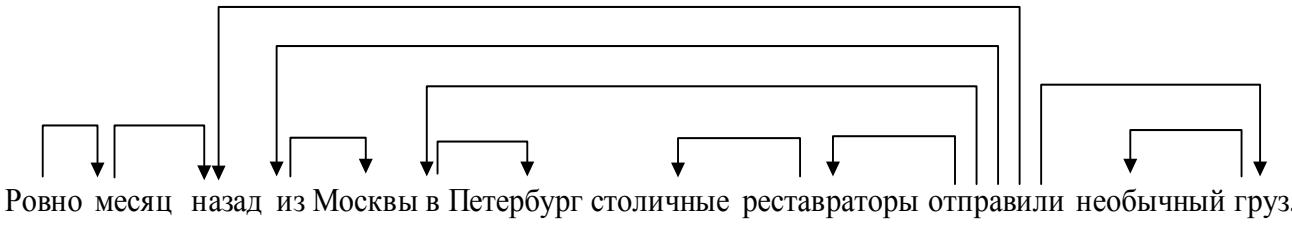


Рисунок 10 – Синтаксическая структура, которая подается на вход семантического анализатора

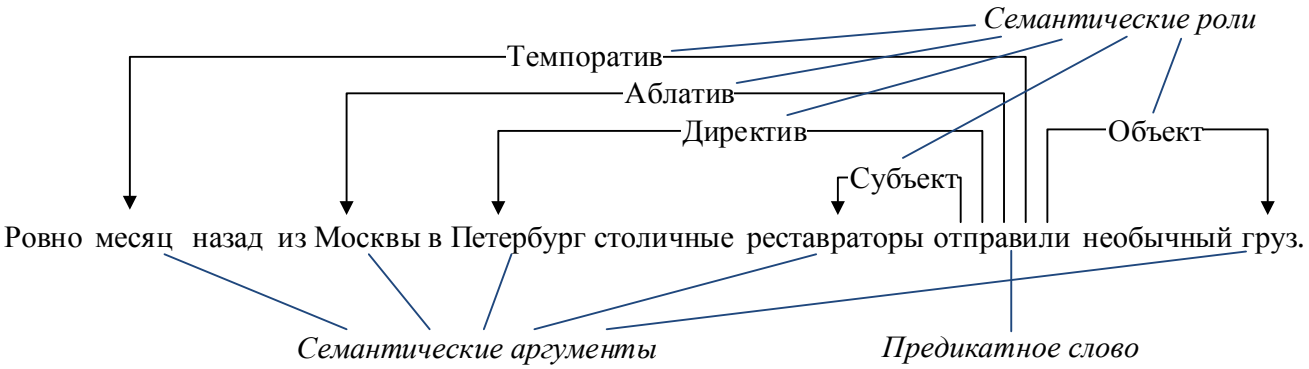


Рисунок 11 – Семантическая структура – результат работы семантического анализатора

2.2. Метод определения ролевых структур высказываний в текстах на русском языке

Основная концепция, представленного в этом разделе метода, состоит в том, что, имея небольшой конечный набор эвристик, детерминированный алгоритм и расширяемый семантический словарь можно построить систему для определения ролевых структур высказываний в текстах из любой предметной области.

2.2.1. Общий алгоритм работы семантического анализатора

Алгоритм семантического анализа предложения можно условно разделить на пять этапов:

- поиск предикатных слов (ПС) и соответствующих им словарных статей;
- поиск аргументов ПС;
- назначение семантических ролей найденным аргументам в соответствии с каждой найденной словарной статьей ПС независимо от других аргументов;
- выбор наилучшего распределения семантических ролей по аргументам для заданной словарной статьи с учетом ограничения единственности ролей путем решения оптимизационной задачи о назначениях;
- выбор наилучшей словарной статьи ПС и соответствующего ей набора семантических ролей.

Кратко опишем каждый этап.

Для поиска ПС в предложении используются морфосинтаксические шаблоны. В основном условия накладываются на часть речи и на положение в синтаксическом дереве. При поиске предикатных слов-существительных в предложении текста используются леммы ПС из семантического словаря. После

того, как ПС найдены, для каждого из них устанавливается список соответствующих им словарных статей в семантическом словаре. Он формируется путем выбора из раздела ролевых структур ПС семантического словаря всех статьей, в которых присутствует лемма главного слова ПС.

Для каждого найденного в предложении ПС ищутся его семантические аргументы. Поиск ведется в соответствии с заданным набором эвристик, которые учитывают характеристики ПС, морфологические характеристики просматриваемых слов в предложении, синтаксическую структуру предложения. С каждым найденным аргументом связывается его вес – число от 0 до 1. Вес аргумента отражает уверенность в том, что найденное слово действительно является аргументом ПС.

Для каждой словарной статьи ПС и для каждого семантического аргумента ПС на основе характеристик аргумента и информации, содержащейся в словарной статье, определяется набор семантических ролей, которыми может обладать заданный аргумент независимо от остальных аргументов. В этой процедуре сравниваются падеж, КСК и предлог аргумента с соответствующими признаками в заданной словарной статье для заданного ПС. В результате нечеткого сопоставления этих признаков на основе эвристик определяется оценка соответствия аргумента семантической роли – вес от 0 до 1.

После того, как каждому аргументу для заданной словарной статьи назначен набор семантических ролей и определены их веса, проводится процедура разрешения неоднозначности распределения ролей по аргументам. В ней семантические роли распределяются между аргументами так, чтобы каждый семантический аргумент имел не более одной роли для заданного ПС, и каждая роль для заданной статьи использовалась бы не более одного раза. При этом покрытие аргументов ролями должно быть наилучшим в соответствии с весами аргументов и ролей. Для этого используется алгоритм оптимизации, основанный

на венгерском методе, который решает задачу о назначениях (задачу линейного целочисленного программирования (ЛЦП)).

После того как оптимальное распределение семантических ролей между аргументами найдено для каждой статьи, выбирается наилучшая словарная статья для ПС. В качестве оценки статьи используется значение целевой функции при оптимальном решении соответствующей задачи о назначениях. В итоге формируется набор семантических ролей, который лучшим образом покрывает семантические аргументы для наиболее подходящей предикатному слову словарной статьи.

Общий алгоритм определения ролевой структуры предложения представлен на рисунке 12.

Рисунок 12 – Общий алгоритм определения ролевой структуры предложения

2.2.2. Поиск предикатных слов

Предикатные слова ищутся с помощью набора морфосинтаксических шаблонов. ПС могут быть либо отдельными словами, либо конструкциями, состоящими из нескольких слов (составное ПС). В последнем случае определяется главное слово в этой конструкции. Рассмотрим набор основных шаблонов для поиска ПС в текстах на русском языке.

- Предикатными словами могут выступать отдельные слова следующих частей речи: глаголы, отглагольные существительные, причастия, краткие причастия, деепричастия, прилагательные и краткие прилагательные. При поиске предикатных слов-существительных их леммы в тексте сравниваются с леммами ПС в семантическом словаре.
- ПС могут представлять собой выражения из двух слов с вспомогательным глаголом вида: <вспомогательный глагол или наречие> + <глагол или краткое прилагательное (главное слово)>. Пример: «*начал строить*»; «*был разбит*».
- ПС могут представлять собой словосочетания из двух слов вида: <предикатив> + <глагол (главное слово)>. Пример: «*можно строить*», «*должно начаться*».
- ПС могут представлять собой конструкции из трех слов вида: <предикатив> + <вспомогательный глагол> + <глагол или краткое прилагательное (главное слово)>. Пример: «можно было построить»; «*должно было начаться*»; «*мог быть сломан*».
- Некоторые конструкции из четырех слов также рассматриваются как единое предикатное слово: <предикатив> + <вспомогательный глагол> + <вспомогательное слово> + <глагол (главное слово)>. Пример: «*можно было начать строить*».

После того как ПС выделены, определяются и фиксируются их свойства, важные для дальнейшего семантического анализа. Если ПС представляет собой выражение, состоящее из нескольких слов, то среди них выбирается главное, которое характеризует всю конструкцию, обозначающее ПС.

Рассмотрим основные свойства, которые определяются для ПС.

- Часть речи.
- Лемма. По лемме ведется поиск словарных статей в семантическом словаре. Для большинства ПС лемма извлекается из морфологического разбора главного слова. Но для кратких прилагательных выбирается лемма омонимичного причастия, либо краткого причастия. Это сделано потому, что семантический словарь не содержит статей для кратких причастий. Как правило, ролевая структура причастий и кратких причастий совпадает. Поэтому для них можно использовать общую статью, но признаки ролей для кратких причастий должны быть специальным образом модифицированы (см. 2.2.4).
- Флаг «возвратность». В семантическом словаре опущены также многие статьи для возвратной формы глагола и статьи для кратких прилагательных. Для обработки этих случаев часто можно использовать статью для невозвратной формы глагола. При этом признаки семантических ролей в этом случае необходимо модифицировать (см. 2.2.4). Флаг возвратность устанавливается ПС, которые представлены в тексте глаголами в возвратной форме, кратким прилагательным и причастиям, а также в некоторых других случаях.

По лемме предикатного слова определяется список его словарных статей в семантическом словаре. При этом одной лемме часто соответствует несколько словарных статей. Это происходит из-за смысловой омонимии ПС. В примерах

«*Вино подходит к обеду*» и «*Гости подошли к ресторану*», предикатное слово «подойти» имеет разное значение. В семантическом словаре этим двум значениям соответствуют две разные словарные статьи, в которых указаны различные ролевые структуры.

В итоге формируется список предикатных слов предложения, для которых зафиксированы вышеуказанные характеристики и определены списки их возможных словарных статей.

2.2.3. Поиск семантических аргументов

Для каждого найденного предикатного слова, определяется множество его семантических аргументов в предложении.

Пусть задано некоторое предложение и в нем – предикатное слово. Все слова в предложении просматриваются слева направо и проверяются на соответствие набору условий. В первую очередь проверяется, что слово обладает всеми морфо-лексическими признаками, чтобы считаться семантическим аргументом. В соответствии с постановкой задачи, семантическими аргументами могут быть существительные, местоимения, вопросительные слова и числительные. Часть речи слова известна из морфологического анализа, проверка на то, что слово является вопросительным выполняется с помощью отдельного словаря.

Далее рассматривается, какое место слово занимает в синтаксической структуре предложения. Был выделен ряд синтаксических конструкций, указывающих на то, что слово является семантическим аргументом.

1) Аргумент синтаксически непосредственно подчинен предикатному слову.
2) Аргумент синтаксически подчинен предлогу, а этот предлог подчинен предикатному слову. Здесь также учитывается, что предлоги могут быть составлены из нескольких слов: «*в результате*», «*вместе с*», «*за счет*» и др. Составные предлоги распознаются в предложении отдельной

процедурой, которая использует словарь подобных конструкций. В случае, когда заданное слово подчинено одному из слов сложного предлога, проверяется, что вершина синтаксической группы такого предлога подчинена предикатному слову.

3) В случае, когда предикатным словом является причастие, в качестве семантического аргумента выделяется также и слово, определяемое причастием.
4) В случае, когда предикатным словом является деепричастие, в качестве семантического аргумента выделяется и подлежащее простого предложения, обстоятельством которого является деепричастный оборот с соответствующим предикатным словом.
5) В случае однородных предикатных слов, аргументы, находящиеся по тексту слева от первого однородного предикатного слова, становятся общими семантическими аргументами для всех предикатных слов однородной группы.
6) Если предикатное слово является составным, то перечисленные правила применяются к каждому слову составного предикатного слова.

Указанные синтаксические конструкции соответствуют формату представления синтаксических деревьев зависимостей корпуса СинТагРус [67]. Однако анализатор для выделения семантических аргументов можно настроить и на другой формат, например, AOT.ru, через включение и отключения разных проверок. Настройка может быть полезна для нивелирования различных ошибок синтаксического анализатора, которые неизбежно возникают на реальных текстах. Например, можно выделять в качестве аргументов не только прямые потомки предикатного слова, но и слова, находящиеся на некотором уровне его поддерева. Анализатор назначает выделенным аргументам веса или «степень уверенности» – число от 0 до 1, в зависимости от того какая эвристика сработала при его выделении, на каком уровне поддерева ПС он находится и какими

морфологическими характеристиками обладает. Это позволяет ранжировать выделенные слова по степени уверенности в том, что они действительно являются аргументами заданного ПС. Например, веса местоимений и вопросительных слов занижаются по сравнению с существительными. Веса задаются при конфигурировании анализатора.

2.2.4. Назначение ролей семантическим аргументам

Пусть имеется ПС, множество его семантических аргументов, задана одна словарная статья ПС.

Процедура назначения ролей семантическим аргументам начинается с назначения «основных» ролей. «Основные» («обязательные» или «центральные») роли – это те роли предикатного слова, которые чаще всего при нем встречаются и без которых описание ситуации, обозначенной ПС, часто является семантически не полным. Назначение основных ролей аргументам осуществляется главным образом на основе признаков, содержащихся в разделе ролевых структур предикатных слов семантического словаря.

Если какой-либо семантический аргумент, является вопросительным словом, и на предыдущем этапе ему роль не была назначена, то запускается отдельная процедура назначения ролей вопросительным словам, которая выполняется на основе информации, содержащейся в разделе вопросительных слов семантического словаря.

Если роль семантического аргумента не определена после предыдущих этапов, то запускается процедура определения периферийных или «необязательных» ролей, которая выполняется на основе информации, содержащейся в разделе периферийных ролей.

Общий алгоритм назначения ролей семантическим аргументам представлен на рисунке 13.

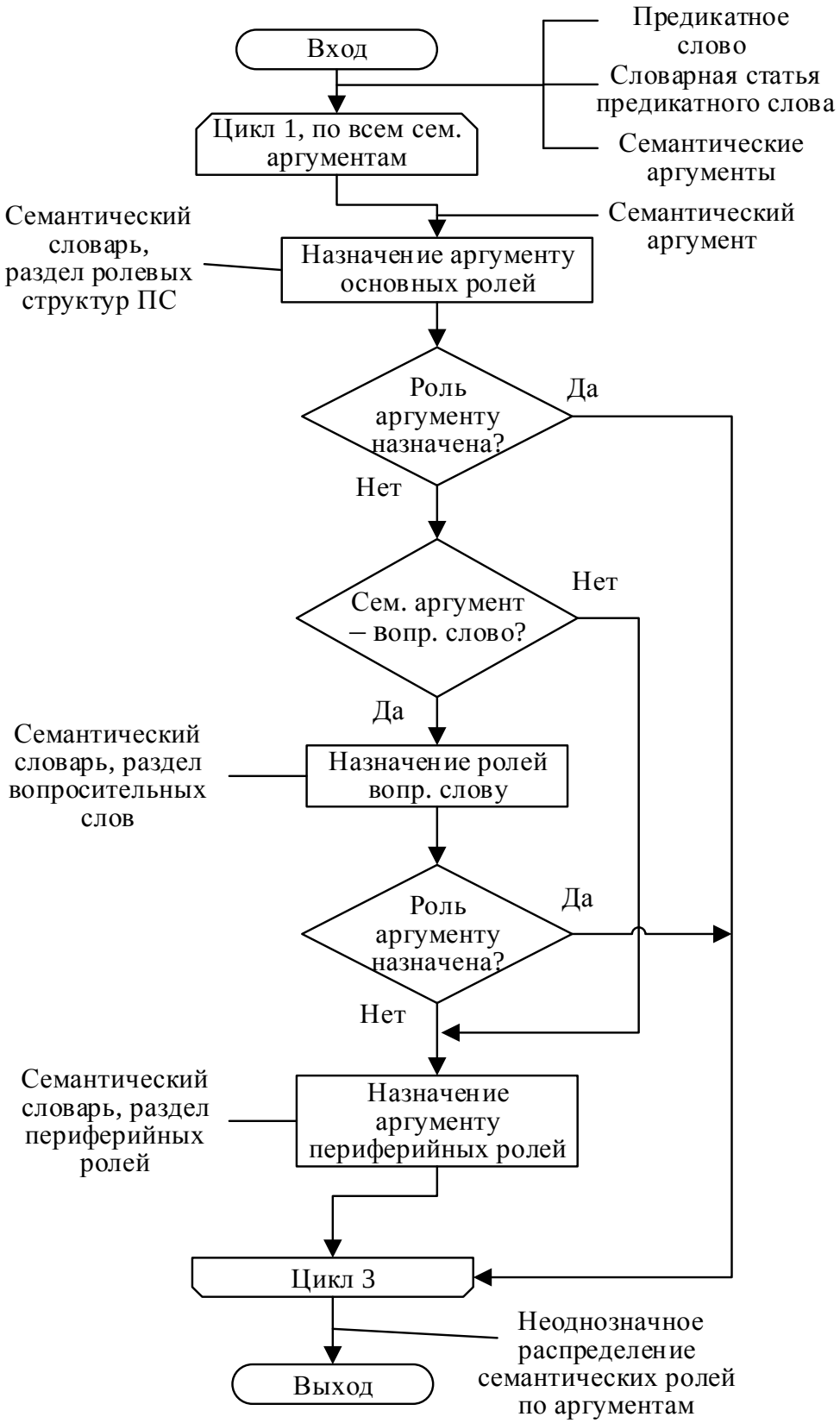


Рисунок 13 – Общий алгоритм назначения ролей семантическим аргументам

Назначение основных ролей аргументам

Пусть задано предикатное слово, одна из его словарных статей и один семантический аргумент.

Все роли заданной словарной статьи просматриваются в цикле и их признаки нечетко сопоставляются с признаками семантического аргумента. Результат сопоставления – число от 0 до 1, которое назовем весом семантической роли. Вес является геометрическим средним двух коэффициентов (от 0 до 1): близости по категориально семантическому классу и близости по грамматическим признакам. Оба коэффициента определяются с помощью набора эвристик.

При определении коэффициента близости по категориально семантическому классу анализатор последовательно сопоставляет все КСК, указанные для заданной роли в словарной статье, с КСК аргумента и для каждого из них определяет оценку сходства. При этом совпадение одних КСК считается весомее совпадения других. На эту оценку также налагается штраф, если у роли в словарной статье указано несколько КСК или роль допускает любой класс. Если значением КСК аргумента является «не определено», то оценка приобретает небольшой значение большее нуля. В качестве итогового коэффициента близости по КСК выбирается максимальная оценка.

Определение коэффициента близости по грамматическим признакам осуществляется путем сопоставления пары признаков (предлог; падеж) семантического аргумента с признаками, указанными для заданной роли в словарной статье. Как правило, в словарной статье указана не одна, а несколько подходящих заданной семантической роли пар признаков. Коэффициенты близости рассчитываются отдельно для каждой пары, а затем из них выбирается максимальный коэффициент и соответствующая ему пара признаков из словаря.

Коэффициент близости по грамматическим признакам рассчитывается следующим образом. Если предлог аргумента не совпадает с предлогом роли,

значение коэффициента – 0. Иначе рассчитывается близость по падежам. Перед расчетом оценки близости по падежам у заданного предикатного слова проверяется установлен ли признак «возвратность». Если этот признак установлен, то падежи из словарной статьи для заданной роли меняются следующим образом. Если предлог для заданной роли отсутствует, то именительный падеж заменяется на творительный, а винительный – на именительный. Существует также ряд менее значимых эвристик преобразования падежа, среди них, например, преобразование именительного падежа в родительный в конструкциях при безличных глаголах с отрицанием (например, «Ответ не пришел → Ответа не пришло»). Некоторые преобразования налагают на коэффициент близости по грамматическим признакам штраф.

Из-за падежной омонимии у аргумента может быть несколько вариантов падежа. Если у аргумента один вариант падежа, который совпадает с падежом семантической роли, то оценка близости по падежам равна 1. Если у аргумента несколько вариантов падежа и, хотя бы один из них совпадает с падежом роли в словарной статье, то оценка не ноль (конкретное значение (0; 1] задается при конфигурировании анализатора). Иначе оценка близости по падежам равна нулю.

После обработки всех ролей, присутствующих в словарной статье, формируется список возможных ролей семантического аргумента и их весов. На рисунке 14 представлен алгоритм назначения ролей аргументу и определения их весов.

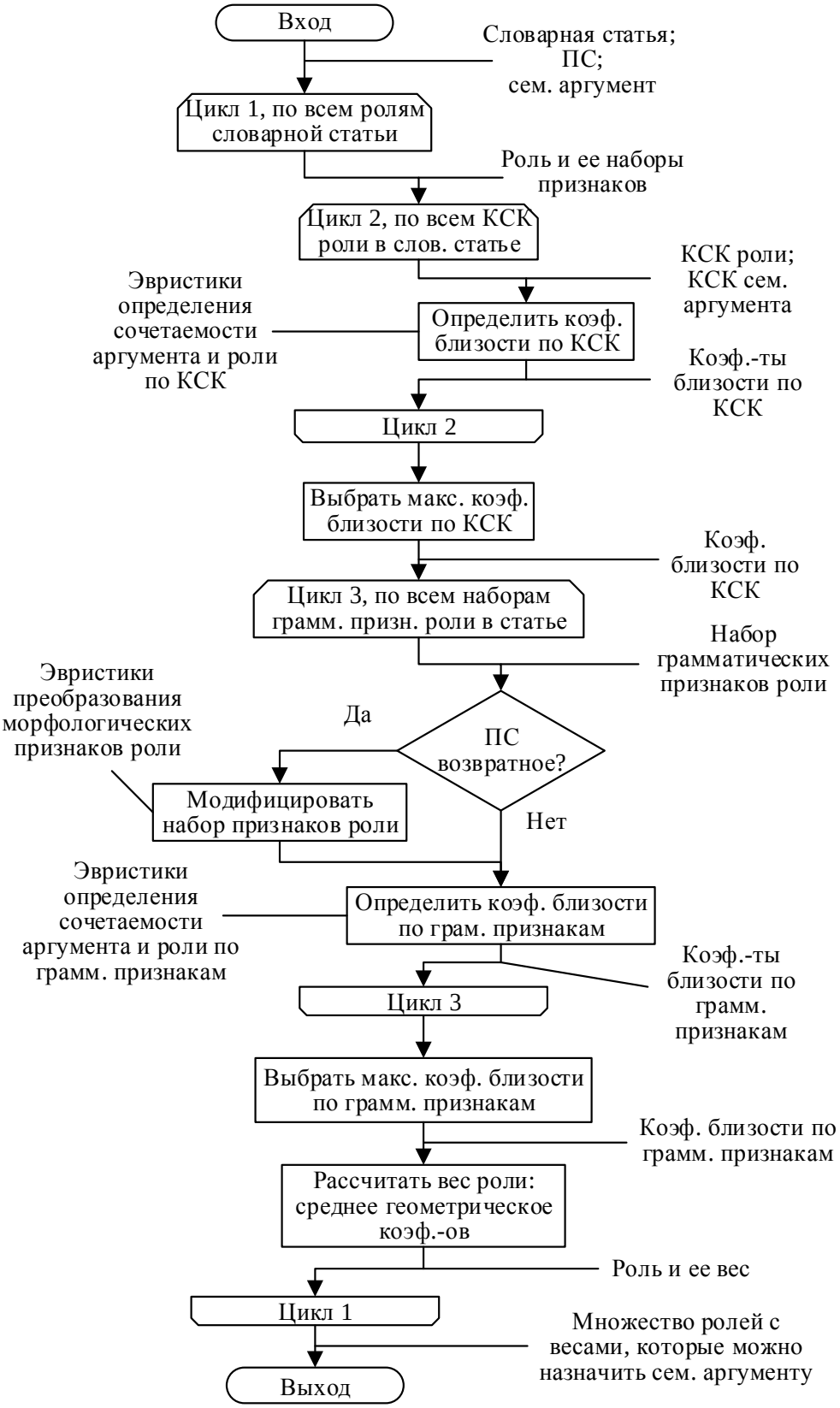


Рисунок 14 – Алгоритм назначения семантической роли аргументу и определения ее веса

Назначение семантических ролей аргументам, выраженными вопросительными словами

В этой процедуре определяются роли семантических аргументов выраженными вопросительными словами: «кто», «что», «когда», «где», «куда» и др. Обрабатываются только те семантические аргументы, роль которым в результате процедуры определения основных ролей не была назначена. Семантические роли вопросительным словам назначаются с помощью отдельного раздела семантического словаря. В этом разделе содержатся словарные статьи, в которых указано, какие роли могут иметь вопросительные слова и какими морфологическими признаками для этого они должны обладать.

По лемме вопросительного слова в словаре ищется советующая ему статья. В случае, если вопросительное слово является наречием или союзом, то все роли из словарной статьи добавляются к списку возможных семантических ролей семантического аргумента. Если нет, то проверяются морфологические признаки вопросительного слова. Морфологические признаки проверяются аналогично тому, как это делается при назначении аргументам основных ролей.

В финальном списке возможных семантических ролей вопросительных слов оставляются только те роли, которые есть в словарной статье предикатного слова и роли, которые входят в список периферийных семантических ролей: локатив, темпоратив, инструментив, директив, каузатив, и др. Веса ролей вопросительных слов занижаются по сравнению с весами основных ролей других семантических аргументов.

Назначение периферийных семантических ролей аргументам

Процедура назначения периферийных ролей обрабатывает только те семантические аргументы, которым после предыдущих процедур ни одна роль еще не назначена. Периферийные семантические роли назначаются аргументам с помощью раздела периферийных ролей семантического словаря.

Для заданного семантического аргумента формируется множество наборов признаков: троек (КСК, падеж, предлог). Эти тройки ищутся в разделе семантического словаря для периферийных ролей, и из него выбираются соответствующие им семантические роли. Для каждой выбранной роли проверяется присутствует ли она в словарной статье предикатного слова аргумента. Если такая роль в статье предикатного слова присутствует, то она отбрасывается. Иначе роль добавляет в список возможных семантических ролей аргумента, ее вес занижается по сравнению с весом основных ролей.

2.2.5. Разрешение неоднозначности назначения семантических ролей аргументам

В результате всех предыдущих этапов для каждого семантического аргумента формируется список нескольких возможных семантических ролей. Причем у разных семантических аргументов роли в этих списках могут повторяться. Однако в соответствии с принципами модели семантики текста, основанной на ролевой структуре предложения, с одной стороны, каждый семантический аргумент может иметь только одну роль, а, с другой стороны, одна и та же роль не может быть у нескольких семантических аргументов. Возникает проблема оптимально распределить семантические роли по аргументам, так, чтобы оценка ролевой структуры для заданного предикатного слова была максимальной. Предлагается рассматривать эту проблему как задачу о назначениях, которая в свою очередь представляет собой задачу линейного целочисленного программирования. В общем случае постановка задачи о назначениях в виде задачи ЛЦП выглядит следующим образом.

$$\begin{gathered}\underset{x_{ij}}{\operatorname{argmin}} \sum_{i,j=1}^{n} C(i,j)x_{ij} \\ \sum_{i=1}^{n} x_{ij} = 1, j = 1, \dots n \\ \sum_{j=1}^{n} x_{ij} = 1, i = 1, \dots n \\ x_{ij} \in \{0,1\}, i, j = 1, \dots n\end{gathered} \tag{4}$$

Здесь x_{ij} – переменные, $C(i,j)$ – функция стоимости.

Представим математическую постановку задачи о назначениях семантических ролей семантическим аргументам. Пусть имеется упорядоченное множество A – семантические аргументы. Пусть R_i – множество возможных ролей аргумента $a_i \in A$. Пусть R – упорядоченное множество всех уникальных ролей, которые можно установить аргументам A. Поскольку в общем случае $|A| \neq |R|$, зададим $n = \max\{|A|, |R|\}$ и два множества A* и R^*, такие, что $|A^*| = |R^*| = n;\ a_i \in A, a_i^* \in A^*, a_i = a_i^*, i = 1, \dots |A|; r_j \in R, r_j^* \in R^*, r_j = r_j^*, j = 1, \dots |R|$. Функцию стоимости $C(a,r)$ зададим как произведение веса аргумента w^a, определенного на этапе поиска семантических аргументов, на вес соответствующей аргументу семантической роли w^r. Причем, если в списке возможных семантических ролей заданного аргумента $a_k \in A$ роль $r_m \in R$ отсутствует ($r_m \notin R_k$), то стоимость равна нулю $C(a_k, r_m) = 0$. Пусть для всех аргументов $a_i^* \in A^*$, которые не входят в исходное множество A ($a_i^*, i > |A|$), функция стоимости равна нулю. Пусть для всех ролей $r_j^* \in R^*$, которые не входят в исходное множество R ($r_j^*, j > |R|$), функция стоимости равна нулю. Запишем полное выражение для функции стоимости:

$$C(a_i, r_j) = \begin{cases} w_i^a * w_j^r, i \le |A|, j \le |R|, r_j \in R_i \\ 0, r_j \notin R_i \\ 0, i > |A| \text{ или } j > |R| \end{cases} \quad (5)$$

Зададим переменные $x_{ij} \in \{0,1\}$, где $x_{ij} = 1$ означает, что аргументу a_i назначена роль r_j. Оценку семантической структуры представим в виде линейного функционала $Sem = \sum_{i,j=1}^{n} C(a_i, r_j) x_{ij}$. Тогда задачу назначения семантических ролей семантическим аргументам можно представить в следующем виде.

$$\begin{gathered} \underset{x_{ij}}{\operatorname{argmax}} Sem = \underset{x_{ij}}{\operatorname{argmax}} \sum_{i,j=1}^{n} C(a_i, r_j) x_{ij} \\ C(a_i, r_j) = \begin{cases} w_i^a * w_j^r, i \le |A|, j \le |R|, r_j \in R_i \\ 0, r_j \in R_i \\ 0, i > |A| \text{ или } j > |R| \end{cases} \\ \sum_{i=1}^{n} x_{ij} = 1, j = 1, \dots n \\ \sum_{j=1}^{n} x_{ij} = 1, i = 1, \dots n \\ x_{ij} \in \{0,1\}, i, j = 1, \dots n \end{gathered} \quad (6)$$

Представленная задача решается с помощью венгерского метода. В соответствии с полученным решением аргументам назначаются роли из заданной словарной статьи ПС. Полученная структура является оптимальной для выбранных оценок ролей и аргументов. Значение целевой функции определяет на сколько хорошо заданная словарная статья подходит для заданного

предикатного слова и его аргументов. Оно используется для выбора оптимальной словарной статьи и соответствующей ей ролевой структуры предикатного слова.

2.3. Метод семантико-синтаксического анализа текстов

Семантические роли могут быть дополнительным источником информации для синтаксического анализа. Они несут в себе знания о сочетаемости предикатных слов с аргументами. Ролевая структура предикатного слова определяет набор «обязательных» аргументов, которые должно присоединить к себе предикатное слово. Кроме того, в ходе установления семантических ролей решается задача разрешения семантической омонимии предикатных слов (выбор словарной статьи). Поэтому семантическая информация оказывается полезной, например, в случаях неоднозначности подчинения именных (или предложных) групп, когда по правилам грамматики группа может быть либо дополнением сказуемого, либо являться определением внутри некоторой другой именной группы. На рисунке 15 представлен случай, в котором синтаксический анализатор потенциально может подчинить предложную группу «без реформ» либо к глаголу «преодолеть», либо к существительному «кризис».

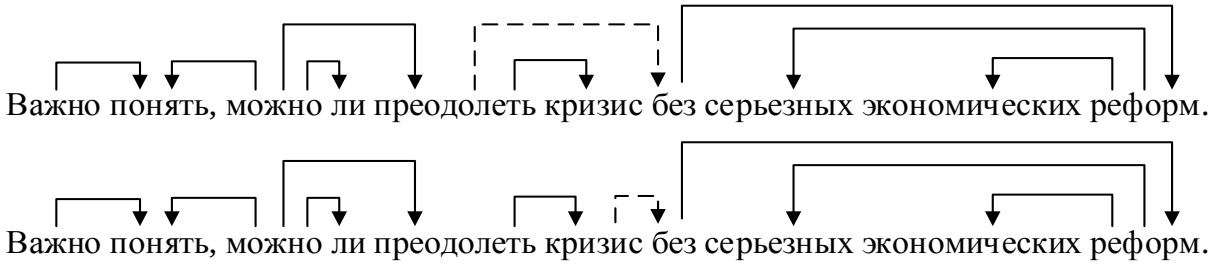


Рисунок 15 – Пример того, как синтаксический анализатор может расставить синтаксические связи в предложении «Важно понять, можно ли преодолеть кризис без серьезных экономических реформ»

Хотя оба варианта грамматически допустимы, только первый вариант синтаксического разбора является семантически верным. Однако без знаний о ролевой структуре предикатного слова «преодолеть», анализатор не может точно разрешить эту неоднозначность.

На рисунке 16 представлен другой пример, в котором наоборот вариант синтаксического разбора, в котором именная группа «без последствий» починена слову «кризис», является семантически верным, а вариант, в котором эта группа связана с глаголом «отразится», – нет.

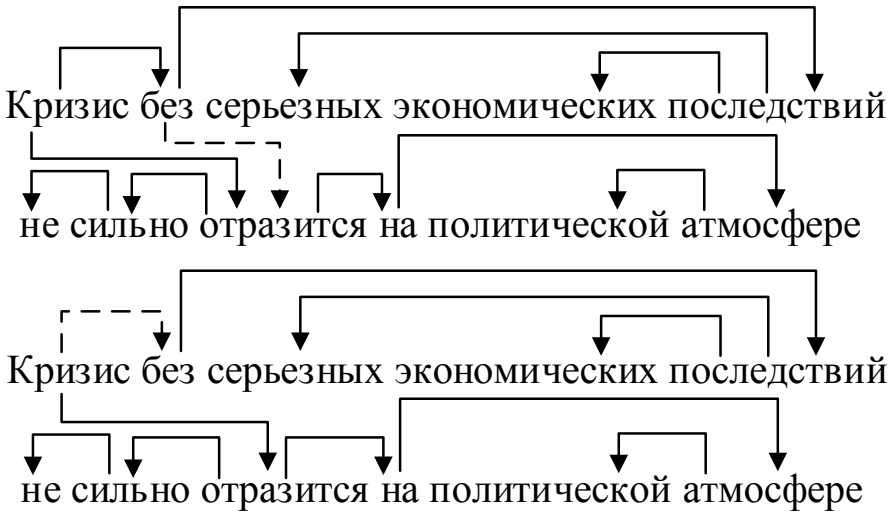


Рисунок 16 – Пример того, как синтаксический анализатор может расставить синтаксические связи в предложении «Кризис без серьезных экономических последствий не сильно отразится на политическую атмосфере»

В этом разделе приводится метод, в котором интегрируются методы построения синтаксических деревьев зависимостей и определения ролевых структур высказываний. За счет знаний о ролевой структуре предложения, полученных от семантического анализатора, метод корректирует исходное синтаксическое дерево зависимостей. Исправления в синтаксическом дереве, в свою очередь, позволяют построить более полные ролевые структуры. Таким образом, метод позволяет повысить качество и синтаксического, и семантического анализа.

2.3.1. Общее описание метода семантико-синтаксического анализа

Предполагается, что перед семантико-синтаксическим анализом проведена токенизация текста, он разбит на предложения, и проведен его морфологический анализ, включая определение морфологических признаков слов, лемматизацию и разрешение омонимии. Семантико-синтаксический анализ проводится для каждого предложения отдельно.

На первом шаге выполняется первоначальный синтаксический анализ предложения. Он может выполняться любым анализатором, который строит синтаксическое дерево зависимостей. В проведенных в работе экспериментах для построения синтаксического дерева использовался анализатор на основе машинного обучения MaltParser. Как было продемонстрировано в нескольких работах [55, 56, 58], MaltParser показывает хорошие результаты при анализе текстов на русском языке. На соревновании синтаксических анализаторов Ru-Eval-2012 он был в тройке лидеров [160]. В разделе 3.2 и в приложении 2 содержатся подробности обучения и настройки синтаксического анализатора. Деревья, построенные в результате первоначального анализа, в дальнейшем корректируются за счет информации, полученной при определении ролевых структур высказываний.

На следующем шаге в предложении ищутся предикатные слова с помощью алгоритма, представленного в разделе 2.2.2. Предикатные слова, выраженные отглагольными существительными или деепричастиями, отбрасываются.

Затем для каждого найденного ПС, исходя из синтаксической структуры предложения определяются его семантические аргументы. Поиск семантических аргументов ПС осуществляется с помощью анализатора, основанного на «сильных» правилах (см. раздел 2.2.3), т.е. такого анализатора, который ищет аргументы строго в соответствии с построенным синтаксическим деревом зависимостей. Если в синтаксическом дереве содержится ошибка, то анализатор не найдет семантический аргумент. Таким образом, набор семантических аргументов, выделенных этим анализатором, является достаточно точным, но может быть неполным. Далее этот набор семантических аргументов будем называть «основным».

Затем ведется поиск конструкций, которые могут быть семантическими аргументами, но из-за потенциальных ошибок в синтаксическом дереве они не были выявлены. Поиск осуществляется с помощью анализатора на основе

«слабых» правил. Этот анализатор не накладывает ограничения на синтаксическую связность предикатного слова и семантического аргумента. Он предполагает, что синтаксические связи могут быть установлены неточно и полагается на другие признаки. Конструкции, выделенные этим анализатором, будем называть «возможными» семантическими аргументами. Набор возможных семантических аргументов является достаточно полным, но неточным. Он, как правило, включает в себя набор основных аргументов.

Среди возможных семантических аргументов, которые не являются основными ищутся «дополнительные» аргументы. Дополнительными аргументами будем называть возможные аргументы, присутствие которых в предложении допускается ролевой структурой предикатного слова, и они обладают для этого соответствующими признаками (морфологическими, категориальными и другими). Дополнительные аргументы фильтруются с помощью набора эвристик.

После того, как все дополнительные аргументы всех предикатных слов предложения найдены, для каждого дополнительного аргумента определяются исправления синтаксического дерева (добавление/удаление связей), которые необходимо совершить, чтобы синтаксическая структура соответствовала новой семантической структуре предложения, с добавленными семантическими аргументами. Исправления также проходят ряд проверок с помощью эристик (проверка на проективность новых синтаксических связей, на отсутствие циклов и др.).

В итоге исправленное синтаксическое дерево заново подается на семантический анализ. Общий алгоритм семантико-синтаксического анализа представлен на рисунке 17.

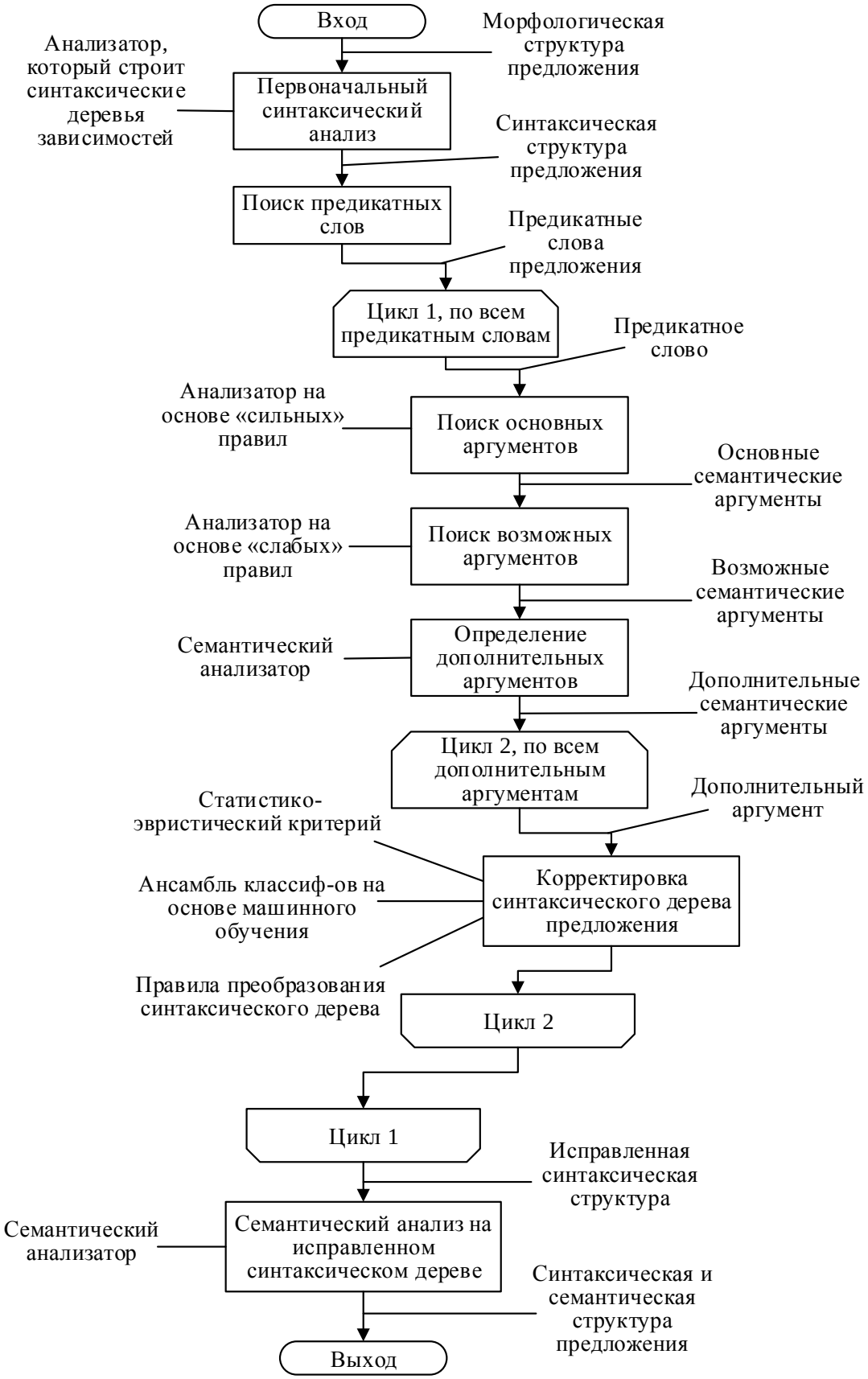


Рисунок 17 – Общий алгоиртм семантико-синтаксического анализа

2.3.2. Поиск возможных семантических аргументов

Метод поиска возможных семантических аргументов схож с методом поиска семантических аргументов из раздела 2.2.3, однако в нем накладываются менее жесткие ограничения на синтаксическую структуру предложения и не требуется, чтобы возможные аргументы были синтаксически связаны с предикатными словами. Однако для того, чтобы отфильтровать наименее подходящие варианты, в нем вводится ряд дополнительных ограничений на аргументы.

В первую очередь формируются левая и правая границы поиска дополнительных аргументов для заданного предикатного слова. Границы определяется в соответствии с небольшим набором правил, которые призваны отфильтровать слова, попадающие в область, в которой в силу лингвистических соображений аргументы, синтаксически связанные с заданным предикатным словом, находиться не могут, или в этой области могут потенциально находится аргументы, синтаксически связанные с другими предикатными словами. Например, в случае с однородными предикатными словами в одной клаузе, выраженными глаголами, семантические аргументы справа от правого однородного глагола, хотя и могут являться семантическими аргументами левого глагола, синтаксически непосредственно связаны только с правым однородным глаголом, поэтому границы расставляются так, чтобы отфильтровать эти аргументы. При построении границ учитываются знаки препинания и союзы, разделяющие однородные предикатные слова.

Затем в рамках построенных границ выделяются все слова, удовлетворяющие признакам именных аргументов: существительные, личные местоимения, местоимения «который», «что», «один», вопросительные слова и др. Отобранные слова проходят через более сложную серию проверок. Среди наиболее значимых проверок следует отметить следующие:

- Проверка того, что возможный семантический аргумент при заданном ПС не является основным аргументом при другом ПС.

- Проверка скобочной структуры: возможный аргумент должен находиться на том же уровне скобочной структуры, что и заданное предикатное слово. Схожая проверка осуществляется и на структуру кавычек.
- Проверка того, что возможный аргумент является вершиной составной именованной сущности. Например, в конструкции ФИО, разобранной как Ф→И→О, в качестве единственного дополнительного аргумента будет выделено слово Ф.
- Проверка того, что возможный аргумент не входит в устойчивое выражение. В тексте выделяются выражения, обозначающие время, период времени, периодичность, стоимость, стоимость в период времени, составные числа, выражения с процентами, единицы измерения и др.
- Проверка на то, что возможный аргумент не является однородным членом предложения, которое является подчиненным другого однородного члена (см. особенности синтаксической разметки корпуса СинТагРус[5]). Однородные слова могут обозначать один и тот же семантический аргумент, при этом подчиненное однородное слово синтаксически не зависит от предикатного слова, поэтому в качестве дополнительного аргумента оно не рассматривается.

После этого из множества возможных аргументов отсеиваются те аргументы, которые входят в множество основных аргументов путем простого вычитания множеств. В результате операции разности строится множество возможных аргументами, которые потенциально могут быть семантическими аргументами предикатного слова, однако в первоначальном синтаксическом дереве отсутствуют необходимые для этого синтаксические связи.

[5] http://www.ruscorpora.ru/instruction-syntax.html

2.3.3. Определение дополнительных семантических аргументов

Дополнительные семантические аргументы отличаются от возможных аргументов тем, что их наличие подкрепляется возможностью включения их в ролевую структуру предикатного слова. Чтобы из множества возможных аргументов выделить дополнительные аргументы, анализатор пытается назначить возможным аргументам семантические роли, которые не конфликтуют с ролями «основных» аргументов.

Для этого сначала анализатор назначает семантические роли «основным» аргументам заданного предикатного слова. Затем каждый возможный аргумент поочередно добавляется ко множеству основных аргументов, и анализатор пытается назначить роли расширенному набору аргументов. Если добавленному возможному аргументу назначить роль не удалось, то он отвергается. Если роль этому аргументу назначена, то проверяется, принадлежит ли эта роль аргументу из основного нерасширенного набора. Если нет, возможный аргумент добавляется ко множеству дополнительных аргументов, иначе возможный аргумент отвергается.

После того, как все возможные аргументы проанализированы, отобранные дополнительные аргументы добавляются ко множеству основных аргументов, и формируется общий набор семантических аргументов. Процедура семантического анализа запускается на этом наборе заново. Те дополнительные аргументы, которые семантическую роль не получили, отбрасываются. Таким образом решается проблема с дополнительными аргументами, роли которых дублируются.

В результате описанной процедуры строится набор дополнительных аргументов, им назначены роли, при этом эти роли не конфликтуют с ролями аргументов из основного набора. Алгоритм построения дополнительных аргументов представлен на рисунке 18.

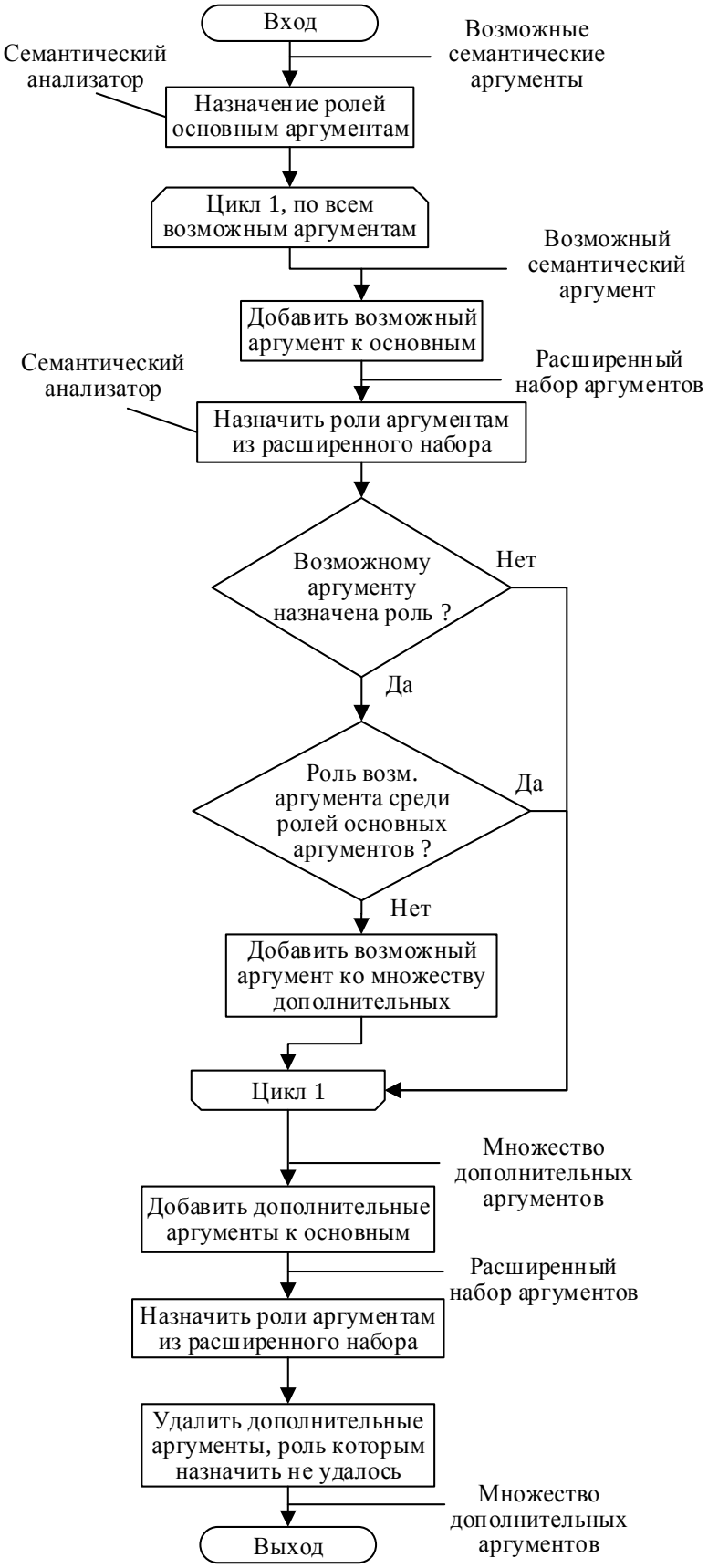


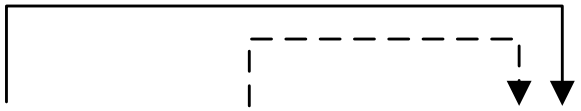
Рисунок 18 – Алгоритм определения дополнительных аргументов

2.3.4. Корректировка синтаксического дерева зависимостей

Для каждого дополнительного аргумента, найденного на предыдущем этапе, проводится попытка откорректировать синтаксическое дерево так, чтобы этот аргумент можно было бы выделить одним из правил алгоритма поиска основных аргументов из раздела 2.2.3. При этом откорректированная синтаксическая структура должна оставаться деревом: каждый узел имеет не более одного родителя, и отсутствуют циклы.

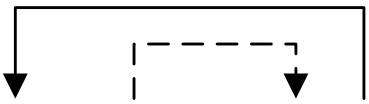
Чтобы исправить дерево, как правило, необходимо удалить из него одну или несколько синтаксических связей и одну связь добавить. Здесь могут возникнуть следующие варианты:

- Слово, обозначенное как дополнительный аргумент, не подчинено предлогу или конструкции, обозначающей составной предлог. В этом случае оно переподчиняется непосредственно предикатному слову, при этом синтаксическая связь между словом и его исходным родителем удаляется. Например, семантический аргумент – существительное подчиняется предикатному слову – глаголу (рисунок 19, А)).
- Слово, обозначенное как дополнительный аргумент, синтаксически подчинено предлогу или конструкции, обозначающей составной предлог. В этом случае переподчиняется не слово, а предлог. В случае составного предлога переподчиняется его синтаксическая вершина (рисунок 19, Б)).
- Предикатное слово является причастием. Тогда отдельный алгоритм определяет может ли дополнительный аргумент быть определяемым словом. Он проверяет согласованность причастия и дополнительного аргумента в роде, числе и падеже, а также определяет имеет ли причастие в качестве альтернативы согласованного родителя. Если аргумент согласован с причастием, и у причастия нет альтернативного согласованного родителя, то причастие переподчиняется этому аргументу (рисунок 19, В)).



А) … произошло ли вызывающее боль смещение межпозвонкового диска

Б) Настоящие балетмейстеры о мимансе как раз помнили.



В) … выделено … на создание не имеющего аналогов суперкомпьютера ...

Рисунок 19 – Примеры способов корректировки синтаксического дерева. Пунктирная стрелка обозначает синтаксическую связь, ошибочно установленную исходым синтаксическим анализатором, которая удаляется системой семантико-синтаксического анализа; сплошная стрелка обозначает новую связь, добавленную системой семантико-синтаксического анализа

После того как определены преобразования, которые необходимо применить к синтаксическому дереву для его корректировки, проверяется, что в результате этих преобразования не возникает цикла, проходящего через дополнительный аргумент и предикатное слово. Если подобный цикл образуется, то он может быть разорван двумя способами:

- Если предикатным словом является причастие, а дополнительный аргумент является определяемым словом, то в цикле удаляется связь, вершиной которой является предикатное слово.
- Иначе в цикле удаляется связь, вершиной которой является аргумент.

Далее проверяется, что добавленная связь в исправленном синтаксическом дереве является проективной. Эта проверка нужна потому, что доля непроективных связей в русском языке составляет всего около 1% [161],

следовательно, нарушение свойства проективности является сигналом того, что связь является ошибочной.

Перед фиксацией исправлений выполняется еще одна проверка преобразований синтаксического дерева, призванная минимизировать ошибочные случаи корректировки. Для этой проверки разработано два метода: с помощью статистико-эвристического критерия, основанного на частоте совместной встречаемости признаков слов; с помощью ансамбля бинарных классификаторов.

Рассмотрим первый метод проверки исправлений с помощью статистико-эвристического критерия. Пусть имеется два существительных между, которыми синтаксическая связь установлена непосредственно, например, «разработка приложений», либо связь установлена через предлог, например, «согласие на разработку». По обучающему корпусу рассчитывается частота совместной встречаемости лемм главного существительного с совокупностью трех признаков зависимого существительного: падеж, предлог и КСК. По небольшой части обучающего корпуса определяется порог. Если при анализе исходный синтаксический анализатор установил между двумя существительными синтаксическую связь, и частота совместной встречаемости леммы главного слова с признаками зависимого выше этого порога, то исправление в синтаксическом дереве не производится.

Во втором методе проверки исправлений используется набор из трех бинарных классификаторов на основе машинного обучения. Первый классификатор C_d оценивает вес удаленной связи между дополнительным аргументом и его первоначальным родителем. Второй классификатор C_a оценивает вес добавленной связи, между аргументом и предикатным словом. Третий классификатор C_f получает на вход значения целевых функций от первых двух и принимает окончательное решение фиксировать или нет изменения в синтаксическом дереве.

Пусть имеется слово – дополнительный аргумент, главное слово, от которого синтаксически зависит аргумент в исходном синтаксическом дереве (эта связь удаляется), и предикатное слово, которому необходимо переподчинить аргумент. Рассмотрим наборы признаков для каждого классификатора («+» означает конкатенацию признаков).

В классификаторе C_d используются следующие признаки:

1) Часть речи главного слова + часть речи аргумента.
2) Часть речи главного слова + падеж аргумента + род аргумента + число аргумента + падеж главного слова + род главного слова + число главного слова.
3) Предлог аргумента + падеж аргумента + часть речи главного слова.
4) Морфологические признаки главного слова + морфологические признаки аргумента + часть речи главного слова.
5) Если главное слово – существительное: лемма главного слова + падеж аргумента + предлог аргумента + КСК аргумента.
6) Если главное слово имеет родителя в исходном синтаксическом дереве: часть речи родителя главного слова + часть речи аргумента + часть речи главного слова.
7) Если родитель главного слова имеет родителя в исходном синтаксическом дереве: часть речи главного слова + часть речи родителя главного слова + часть речи родителя родителя главного слова.
8) Если аргумент – не последнее слово в предложении: часть речи слова, следующего за аргументом, часть речи аргумента, часть речи главного слова.
9) Позиция аргумента относительно главного слова + предлог аргумента + часть речи главного слова.
10) Расстояние в токенах между главным словом и аргументом.

В классификаторе C_a используются следующие признаки:

1) Лемма предикатного слова + КСК аргумента + падеж аргумента + предлог аргумента + позиция аргумента относительно предикатного слова + флаг возвратности предикатного слова.
2) Расстояние между аргументом и предикатным словом в токенах.
3) Падеж аргумента + КСК аргумента + позиция аргумента относительно предикатного слова + предлог аргумента.
4) Если у предикатного слова есть родитель: часть речи родителя предикатного слова.
5) Флаг, сигнализирующий о том, что аргумент и предикатное слово находятся в одной клаузе.

Классификатор C_f использует следующие признаки:

1) Значение целевой функции классификатора C_d.
2) Значение целевой функции классификатора C_a.
3) Семантическую роль аргумента.
4) Вес семантической роли.
5) Падеж аргумента.

Признаковые пространства классификаторов C_d и C_a весьма велики. Поэтому, в угоду скорости семантико-синтаксического анализа применялся один из самых быстрых дискриминантных алгоритмов классификации – метод опорных векторов без ядра (с линейным ядром). В классификаторе C_f гораздо меньше признаков. Проведенные эксперименты показали, что лучшее качество проверки исправлений дает метод опорных векторов с радиальным ядром.

Расчет частот совместной встречаемости признаков для первого метода проверки исправлений и обучение классификаторов для второго метода проводилось на части корпуса СинТагРус.

При обучении классификатора C_d положительные и отрицательные примеры извлекались из корпуса следующим образом:

- В качестве положительных примеров рассматривались пары слов, связанные синтаксической связью, где зависимое слово имеет признаки семантического аргумента, а главное слово не выделено как предикатное слово.
- В качестве отрицательных примеров рассматривались пары слов, между которыми отсутствует синтаксическая связь, при этом одно из них имеет признаки семантического аргумента, и связь, которая может быть проведена между ними, является проективной по отношению ко всем связям синтаксического дерева за исключением той связи, в которой слово, имеющее признаки аргумента, является зависимым.

При обучении классификатора C_a положительные и отрицательные примеры извлекались из корпуса следующим образом:

- В качестве положительных примеров рассматривались пары слов, между которыми присутствует синтаксическая связь, при этом одно из них является предикатным словом, а другое имеет признаки семантического аргумента.
- В качестве отрицательных примеров рассматривались пары слов, между которыми отсутствует синтаксическая связь, при этом одно из них является предикатным словом, а другое имеет признаки семантического аргумента, и связь, которая может быть проведена между ними, является проективной по отношению ко всем связям синтаксического дерева за исключением той связи, в которой слово, имеющее признаки аргумента, является зависимым.

Классификатор C_f обучался на тех случаях исправления синтаксического дерева, которые прошли все предварительные фильтры, включая проверку на проективность результирующего дерева.

После того как все проверки пройдены, исправление фиксируется в синтаксическом дереве и алгоритм переходит к обработке следующего дополнительного аргумента.

Алгоритм корректировки синтаксического дерева зависимостей представлен на рисунке 20.

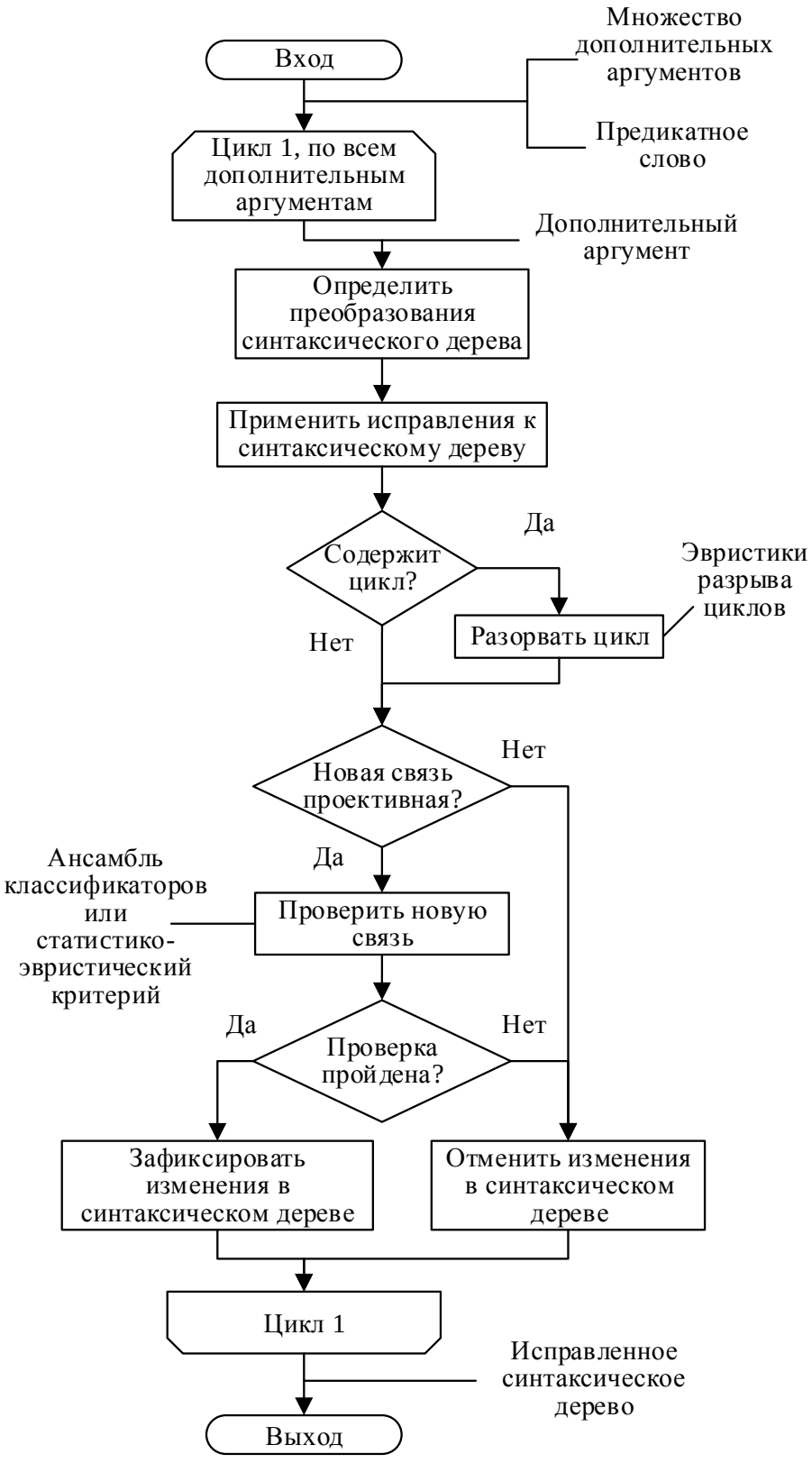


Рисунок 20 – Алгоритм корректировки синтаксического дерева зависимостей

2.4. Выводы

Разработаны новые методы семантического и семантико-синтаксического анализа текстов на естественном языке.

Метод семантического анализа использует семантический словарь в качестве основного источника знаний о предикатных словах и их ролевых структурах. Для поиска предикатных слов используются морфо-лексические шаблоны. Для поиска семантических аргументов предикатного слова используются правила, учитывающие морфологические признаки слов и синтаксическую структуру предложения. Метод назначения ролей аргументам осуществляет нечеткое сравнение признаков аргументов с признаками ролей из статей семантического словаря для заданного предикатного слова. Разрешение неоднозначности распределения ролей по аргументам осуществляется путем решения оптимизационной задачи линейного целочисленного программирования венгерским методом.

Метод семантико-синтаксического анализа использует информацию, полученную в результате семантического анализа для корректировки синтаксического дерева. В этом методе ищутся слова, которые потенциально могут быть семантическими аргументами, но из-за ошибок в синтаксическом дереве предложения они обычным способом не выявляются, и проверяется возможность включить их в ролевую структуру предикатного слова. Если ролевая структура предикатного слова допускает дополнительный аргумент, то синтаксическое дерево корректируется так, чтобы оно соответствовало новой ролевой структуре. При корректировке дерева исключаются потенциально возникающие циклы, проверяется проективность добавленных связей, а также осуществляется проверка преобразования либо с помощью статистико-эвристического критерия, либо с помощью ансамбля бинарных классификаторов на основе машинного обучения. На исправленном синтаксическом дереве заново проводится семантический анализ.

ГЛАВА 3. ЭКСПЕРИМЕНТАЛЬНОЕ ИССЛЕДОВАНИЕ МЕТОДОВ СЕМАНТИЧЕСКОГО И СЕМАНТИКО-СИНТАКСИЧЕСКОГО АНАЛИЗА

3.1. Тестовые данные и оценки качества

Тестирование разработанных методов семантического и семантико-синтаксического анализа проводилось на корпусе СинТагРус [162, 67]. СинТагРус – это синтаксически размеченный корпус текстов на русском языке (банк синтаксических деревьев), разработанный в ИППИ РАН. Корпус проверен вручную экспертами, в нем снята морфологическая и синтаксическая омонимия, поэтому он может выступать в качестве золотого стандарта и в качестве обучающего корпуса для синтаксических анализаторов.

В СинТагРус тексты разбиты на предложения, а предложения на токены. Для токенов указаны форма слова, лемма, часть речи и другие морфологические характеристики. Синтаксическая структура предложения представляет собой дерево синтаксических зависимостей. Синтаксические связи между токенами предложения имеют метки, определяющие тип связи, и образуют связное синтаксическое дерево. Пунктуация в корпусе является «висящей», т.е. не соединяется связями с другими токенами. Версия корпуса, которая использовалась в работе, содержит 53 439 предложений и 774 373 токенов без учета пунктуации. СинТагРус был преобразован в формат CoNLL-2006/2007 [163], а морфологические характеристики токенов были преобразованы в формат, соответствующий стандарту MULTEXT-East Morphosyntactic Specifications, Version 4 (MTE) [164, 165]. На рисунке 21 представлен пример синтаксической

разметки корпуса СинТагРус. Подробная информация о разметке указана в сайте корпуса[6].

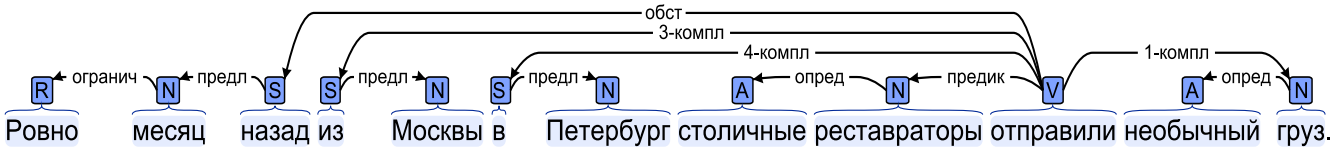


Рисунок 21 – Пример синтаксической разметки корпуса СинТагРус

Для оценки качества решения задачи определения ролевых структур высказываний создан семантический корпус. Для этого вручную был размечен подкорпус СинТагРус. В нем размечены предикатные слова, определены их семантические аргументы и указаны их семантические роли. В подкорпусе также размечены категориально-семантические классы существительных, для которых указана семантическая роль. Отметим, что размечены предикатные слова и роли только для тех случаев, которые описаны семантическим словарем. Весь семантический корпус содержит 1 730 предложений, около 29 000 токенов без учета пунктуации. В корпусе размечено около 3 000 предикатных слов и 4 000 ролей. На рисунке 22 представлен пример разметки семантического корпуса.

Рисунок 22 – Пример разметки предикатных слов, аргументов и ролей в семантическом корпусе

Для оценки качества выделения семантических аргументов был размечен еще один подкорпус размером 200 предложений (около 1 700 токенов без учета пунктуации). В этом подкорпусе для каждого предикатного слова были

[6] http://www.ruscorpora.ru/instruction-syntax.html

размечены все семантические аргументы. Всего размечено около 800 семантических аргументов для более чем 460 предикатных слов.

При тестировании многих методов в этой главе используются следующие метрики: точность p (precision), полнота r (recall) и F_1-мера. Они рассчитываются следующим образом:

$$p = \frac{tp}{tp + fp};\ r = \frac{tp}{tp + fn};\ F_1 = \frac{2 * p * r}{(p + r)}. \quad (7)$$

В общем случае tp (true positives) – количество правильно распознанных положительных примеров; fp (false positives) – количество примеров, которые были ошибочно отнесены к положительным; fn (false negatives) – количество положительных примеров, которые были ошибочно отнесены к отрицательным. В каждом конкретном случае оговаривается, что подразумевается под tp, fp, fn.

3.2. Оценка качества синтаксического анализа, определения категориально-семантических классов, морфологического анализа

Во всех проведенных экспериментах применялся синтаксический анализатор на основе MaltParser, который, как известно из работ [56, 55, 161], является одним из лучших обучаемых синтаксических анализаторов текстов на русском языке. Для его обучения использовалось около 48 000 предложений из корпуса СинТагРус суммарным объемом около 700 000 токенов без учета пунктуации. Основная конфигурация анализатора была взята из работ [55, 56]. В обучающем корпусе синтаксические связи помечались тремя типами меток: метки, обозначающие виды сочинительных связей, метка, указывающая на то, что заданная вершина связана с виртуальной корневой вершиной «ROOT» и

метка по умолчанию. Большое количество синтаксических меток связей многократно замедляет процедуру синтаксического анализа и повышает требования к оперативной памяти. Ограниченный набор меток, позволяет нивелировать эту проблему. Эксперименты по обучению MaltParser на разных конфигурациях и наборах признаков, а также на разных по размеру корпусах приведены в приложении 2.

Для тестирования использовался подкорпус размером 5 000 предложений, содержащий около 70 000 токенов без учета пунктуации.

Поскольку для методов семантического анализа, предлагаемых в работе, приоритетно качество установления связей в синтаксическом дереве, а типы синтаксических связей играют малую роль, в качестве основного критерия оценки качества синтаксического анализа использовался UAS (unlabeled attachment score). UAS рассчитывается как доля правильно установленных синтаксических связей среди всех связей золотого стандарта. Во всех экспериментах UAS определялся без учета пунктуации. Характеристики синтаксического анализатора, который использовался в настоящей работе приведены в таблице 2. Тип морфологических признаков указывает, были ли они взяты из золотого стандарта, или же они автоматически получены от морфологического анализатора (в этом случае они содержат некоторую долю ошибок).

Таблица 2 – Оценки качества работы синтаксического анализатора

Тип морфологических признаков	**UAS**
Признаки золотого стандарта	86,5
Автоматически сгенерированные признаки	84,6

В качестве морфологического анализатора в работе использовался модуль из открытого программного пакета AOT.ru [166]. Этот модуль решает широкий круг задач по графематическому и морфологическому анализу текста:

осуществляет разбиение текста на токены и предложения, проводит лемматизацию слов (как словарную, так и несловарную), определяет их морфологические признаки, а также разрешает морфологическую омонимию. В модуле реализована скрытая Марковская модель, которая позволяет разрешать частеречную и лексическую виды омонимии, а также омонимию полных морфологических признаков (для включения последней в модуле AOT.ru, необходима модификация исходной реализации). В работе [167] авторы приводят точность работы их модуля при разрешении морфологической омонимии: 97,26 % – для частеречной омонимии; 99,17 % – для лексической омонимии; 94,46 % – при разрешения омонимии полных морфологических признаков.

Категориально-семантические классы слов в проведенных экспериментах определялись с помощью отдельного модуля, в основе которого лежит набор словарей, а также эвристики определения КСК по аффиксам слов. Оценка этого модуля проводилась на части семантически размеченного подкорпуса СинТагРус. Использовалось 1 500 семантически размеченных предложений, суммарным объемом около 25 000 токенов без учета пунктуации. Оценивалась точность определения КСК, которая рассчитывалась как отношение количества слов, которым КСК был назначен верно, и количества слов, имеющих КСК-пометки в корпусе. Точность составила 93,7 %.

3.3. Экспериментальное исследование метода поиска семантических аргументов

Метод поиска семантических аргументов тестировался при следующих условиях (конфигурациях):

1) На вход анализатора подаются морфологические признаки и синтаксическая разметка из золотого стандарта. Обозначим такую конфигурацию как «Gold_Morph_Synt».

2) На вход анализатора подаются морфологические признаки и синтаксическая разметка, сгенерированные автоматически. Обозначим такую конфигурацию как «Morph_Synt». Эта конфигурация отражает реальную задачу анализа текста в некоторой прикладной системе обработки текстов на ЕЯ.

Для тестирования метода поиска семантических аргументов использовался размеченный корпус из 200 предложений, в котором для каждого предикатного слова были размечены все его семантические аргументы. Рассчитывались точность p, полнота r и F_1 мера. В этих оценках:

- tp – количество семантических аргументов в выдаче анализатора, совпавших с семантическими аргументами в проверочном корпусе, у которых также совпали позиции ПС в предложении;
- fp – количество семантических аргументов в выдаче анализатора, которые отсутствуют в проверочном корпусе, или у них отличаются позиции ПС;
- fn – количество семантических аргументов в проверочном корпусе, которые отсутствуют в выдаче анализатора, или у них не совпадают позиции ПС.

Результаты представлены в таблице 3.

Таблица 3 – Результаты экспериментальной оценки качества поиска семантических аргументов

Конфигурация	$\boldsymbol{p}$, %	$\boldsymbol{r}$, %	$\boldsymbol{F_1}$, %
Gold_Morph_Synt	94,4	95,0	94,6
Morph_Synt	86,1	83,2	84,6

Результаты иллюстрируют, что разработанный эвристический метод нахождения семантических аргументов при идеальной синтаксической разметке,

как и ожидалось, работает с высокой точностью и полнотой. Это позволяет его использовать для нахождения основных семантических аргументов в системе семантико-синтаксического анализа. Ошибки чаще всего возникают, если предикатное слово и его аргумент связаны через достаточно сложные синтаксические конструкции, например, в случаях, когда в одной клаузе несколько однородных предикатных слов. Качество поиска семантических аргументов сильно снижается при использовании автоматически сгенерированной синтаксической разметки – приблизительно на 10% по F_1-мере. Эксперимент показывает важность качественного синтаксического анализа как для решения задачи поиска семантических аргументов, так и для всей процедуры семантического анализа. Стоит отметить, что в соответствии с результатами, представленными в разделе 3.2, снижение F_1-меры меньше, чем доля ошибок синтаксического анализатора по UAS в сравнении с золотым стандартом. Причиной этому может быть то, что синтаксические связи между предикатными словами и их аргументами устанавливаются точнее, чем в целом по тексту.

3.4. Экспериментальное исследования метода семантического анализа

Семантический анализатор тестировался при следующих условиях (конфигурациях):

1) На вход анализатора подаются морфологические признаки, синтаксическая разметка и КСК из золотого стандарта. Обозначим такую конфигурацию как «Gold_Morph_CSC_Synt».
2) На вход анализатора подаются морфологические признаки и синтаксическая разметка из золотого стандарта, а КСК определяются автоматически. Обозначим такую конфигурацию как «CSC_Gold_Morph_Synt».

3) На вход анализатора подаются морфологические признаки и КСК из золотого стандарта, а синтаксическая разметка генерируется автоматически с помощью синтаксического анализатора. Обозначим такую конфигурацию как «Synt_Gold_Morph_CSC».
4) На вход анализатора подаются морфологические признаки из золотого стандарта, а КСК и синтаксическая разметка генерируются автоматически анализаторами. В этой конфигурации используется модель для синтаксического анализатора, обученная на морфологических признаках из золотого стандарта. Обозначим такую конфигурацию как «CSC_Synt_Gold_Morph».
5) Все признаки и вся разметка, подаваемая на вход семантического анализатора, генерируются автоматически. Эта конфигурация соответствует реальной практической задаче анализа текста: на вход анализатора подается просто текст, на выходе строится синтаксическое дерево и ролевая структура. Обозначим такую конфигурацию как «Morph_CSC_Synt».

Для тестирования анализатора для определения ролевых структур высказываний было взято около 1 500 предложений, что составляет примерно 25 000 токенов без учета пунктуации. В тестовом подкорпусе размечено около 3 300 ролей. Остальная часть семантически размеченного корпуса использовалась для настройки анализатора.

Рассчитывались точность p, полнота r и F_1-мера. При сопоставлении результатов анализатора с тестовым подкорпусом учитывались только те семантические роли, которые присутствуют в тестовом подкорпусе. В этом случае:

- tp – количество аргументов, роль которых при заданном ПС совпадает с ролью, указанной в тестовом подкорпусе при этом ПС;

- fn – количество аргументов, роль которых при заданном ПС указана в тестовом подкорпусе, но в выдаче анализатора роль для этого аргумента при заданном ПС не определена;
- fp – количество аргументов, роль которых при заданном ПС указана в тестовом подкорпусе, в выдаче анализатора роль для этого аргумента при заданном ПС определена, но она не совпадает с ролью, указанной в подкорпусе.

Результаты оценки качества решения задачи определения ролевых структур высказываний представлены в таблице 4.

Таблица 4 – Результаты экспериментальной оценки качества семантического анализатора

Конфигурация	***p*, %**	***r*, %**	***F₁*, %**
Gold_Morph_CSC_Synt	92,3	80,8	86,2
CSC_Gold_Morph_Synt	89,4	73,4	80,6
Synt_Gold_Morph_CSC	92,1	74,3	82,3
CSC_Synt_Gold_Morph	89,2	67,3	76,7
Morph_CSC_Synt	89,6	61,0	72,6

Полученные результаты свидетельствуют о том, что разработанный семантический анализатор для анализа текстов на русском языке демонстрирует качество решения задачи определения ролевых структур высказываний на уровне современных анализаторов для других языков с малым количеством языковых ресурсов (см. в разделе 1.4, испанский, чешский, китайский и др.). Исходя из полученных результатов можно судить о важности различных признаков для решения этой задачи. В первую очередь необходимо отметить большое влияние на конечный результат ошибок при определении категориально-семантических классов. При переходе от конфигурации «Gold_Morph_CSC_Synt» к «CSC_Gold_Morph_Synt» значительно снижается как точность (около 3%), так и

полнота (около 7,5%). Для повышения точности определения КСК слов можно использовать методы кластеризации больших коллекций, а также лингвистические тезаурусы. Использование автоматически сгенерированной разметки в основном ведет лишь к снижению полноты: при переходе от конфигурации «Gold_Morph_CSC_Synt» к конфигурации «Synt_Gold_Morph_CSC» полнота снижается примерно на 6,5%. Это является отражением ошибок, возникающих на этапе выявления семантических аргументов. При переходе от конфигураций, в которых морфологические признаки берутся из золотого стандарта, к конфигурации, в которой морфологические признаки генерируются автоматически («Morph_CSC_Synt»), накладываются как ошибки сравнения морфологических признаков ролей и аргументов, так и снижение точности синтаксического анализа (приблизительно на 2% по разделу 3.2).

Как видно из результатов, только примерно половина всех ошибок при определении ролевых структур высказываний обусловлена ошибками на предыдущих этапах обработки текста: морфологическом, синтаксическом анализе и определении КСК. В другой половине значимую роль играют ошибки непосредственно методов, применяемых для определения ролевых структур высказываний. Как видно из результатов, представленных в разделе 3.3, метод поиска аргументов не дает стопроцентной точности из-за множества различных сложных случаев предикатно-аргументных связей. Еще одной проблемой семантического анализатора, которая обуславливает значительную долю ошибок, является ограничения семантического словаря. В словарных статьях семантического словаря содержатся всего три признака для дифференциации семантических ролей, которых иногда недостаточно. Существует значительное количество случаев, в которых по этим признакам нельзя отделить две семантические роли. Частично эта проблема решается эвристиками. Еще одним ограничением семантического словаря являются содержащиеся в нем ошибки.

На рисунках 23 и 24 представлены примеры семантического анализа предложений из корпуса СинТагРус. Зеленым цветом обозначены предикатные слова, красным цветом – аргументы. На красных стрелках указаны семантические роли, зеленые стрелки обозначают семантические отношения в семантической сети (см. раздел 4.1).

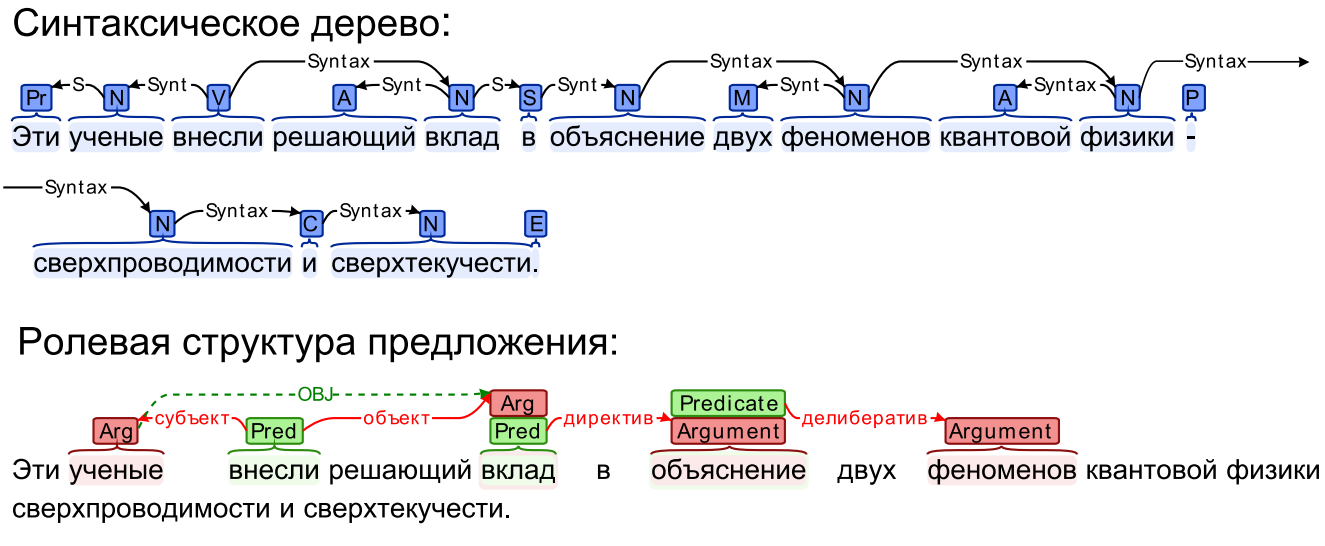


Рисунок 23 – Результат семантического анализа предложения из корпуса СинТагРус «Эти ученые внесли решающий вклад в объяснение двух феноменов квантовой физики – сверхпроводимости и сверхтекучести.»

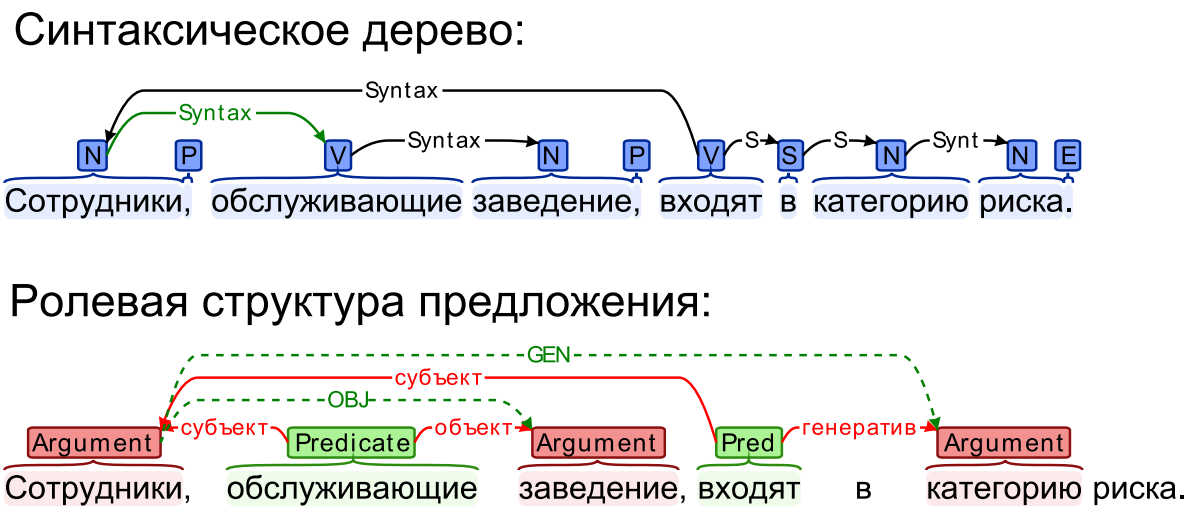


Рисунок 24 – Результат сементического анализа предложения из корпуса СинТагРус «Сотрудники, обслуживающие заведение, входят в категорию риска.»

3.5. Экспериментальное исследование метода семантико-синтаксического анализа

В работе проводилась оценка влияния разработанного метода семантико-синтаксического анализа на качество построения синтаксических деревьев зависимостей и определения ролевых структур высказываний.

3.5.1. Оценка влияния метода семантико-синтаксического анализа на качество синтаксического анализа

Для того, чтобы оценить влияние разработанного метода на качество построения синтаксических деревьев зависимостей, система семантико-синтаксического анализа сравнивалась с синтаксическим анализатором, реализованным с помощью MaltParser. Он также использовался для предварительного синтаксического анализа в системе семантико-синтаксического анализа. Тестирование и обучение проводилось на корпусе СинТагРус, который был разделен на два подкорпуса. В обучающий подкорпус

входит около 48 тыс. предложений и 700 тыс. токенов без учета пунктуации. В тестовый подкорпус входит 5 тыс. предложений и более 70 тыс. токенов без учета пунктуации. Первый подкорпус использовался для обучения синтаксического анализатора на основе MaltParser (см. раздел 3.2 и приложение 2). На нем же рассчитывались частоты совместной встречаемости признаков слов для статистико-эвристического критерия, обучались классификаторы C_f, C_a, C_d, которые проверяют исправления синтаксической структуры перед их фиксацией, а также при помощи перекрестной проверки настраивались их параметры.

На тестовом подкорпусе проводилась оценка качества установления синтаксических связей между ПС и семантическими аргументами. Это означает, что при оценке учитывались только синтаксические связи, где главным словом является глагол, отглагольное существительное, краткое прилагательное, причастие, деепричастие, а зависимым – существительное, местоимение, предлог, которому подчинено существительное или местоимение, вопросительное слово, а также связи, где главное – существительное или местоимение, а зависимое – причастие. В качестве оценочных метрик рассчитывались точность p, полнота r и F_1-мера. Здесь:

- tp – количество синтаксических связей между ПС и их аргументами в тестовом подкорпусе, совпавших со связями в выдаче анализатора;
- fp – количество связей в выдаче анализатора, установленных между ПС и словами, имеющими признаки аргумента, которые отсутствуют в тестовом подкорпусе;
- fn – количество связей между ПС и их аргументами, которые присутствуют в тестовом подкорпусе, но отсутствуют в выдаче анализатора.

В экспериментах сравнивались два способа проверки исправлений в семантико-синтаксическом анализе: с помощью статистико-эвристического критерия, основанного на проверке частоты совместной встречаемости

признаков слов, и критерия на основе ансамбля бинарных классификаторов, использующих машинное обучение: C_f, C_a, C_d. Таким образом, сравнивались три анализатора:

- синтаксический анализатор на основе MaltParser, обозначим его как «MaltParser»;
- система семантико-синтаксического анализа, в которой реализован статистико-эвристический критерий проверки исправлений, обозначим ее как «Сем.-син. Стат.»;
- система семантико-синтаксического анализа, в которой реализован критерий проверки исправлений с помощью классификаторов на основе машинного обучения, обозначим его как «Сем.-син. МО».

Результаты экспериментальных исследований этих анализаторов на тестовом синтаксически размеченном корпусе представлены в таблице 5.

Таблица 5 – Сравнение качества синтаксического анализа системы семантико-синтаксического анализа с синтаксическим анализатором MaltParser

Анализатор	$\boldsymbol{p}$, %	$\boldsymbol{r}$, %	$\boldsymbol{F_1}$, %
MaltParser	**90,6**	89,0	89,8
Сем.-син. Стат.	90,5	**90,2**	90,3
Сем.-син. МО	**90,6**	90,1	**90,4**

Полученные результаты показывают, что разработанный метод семантико-синтаксического анализа позволяет повысить полноту установления синтаксических связей между ПС и аргументами более чем на 1,1 %, тем самым ему удается установить более 10 % всех наиболее сложных синтаксических связей между предикатными словами и аргументами, которые не были найдены MaltParser. При этом точность системы семантико-синтаксического анализа не изменяется. Отсутствие значимого повышения точности объясняется тем, что в полученных оценках учитываются только новые связи, добавленные в

синтаксическое дерево, но не учитываются удаленные. Повышение полноты установления синтаксических связей оказывает значительное влияние на повышение F_1 меры.

Проверялась статистическая значимость полученных результатов. Для этого использовался метод статистического бутстрэппинга (statistical bootstrapping), как это было предложено в работе [168]. Показано, что прирост полноты и F_1-меры системы «Сем.-син. МО» по сравнению с MaltParser статистически значим. Доверительный интервал при уровне значимости $\alpha = 0{,}05$ для полноты составил (0,9; 1,2) %. На рисунке 25 проиллюстрировано распределение прироста полноты (Δr) системы семантико-синтаксического анализа по сравнению с системой, в которой синтаксический и семантический анализ выполняются раздельно.

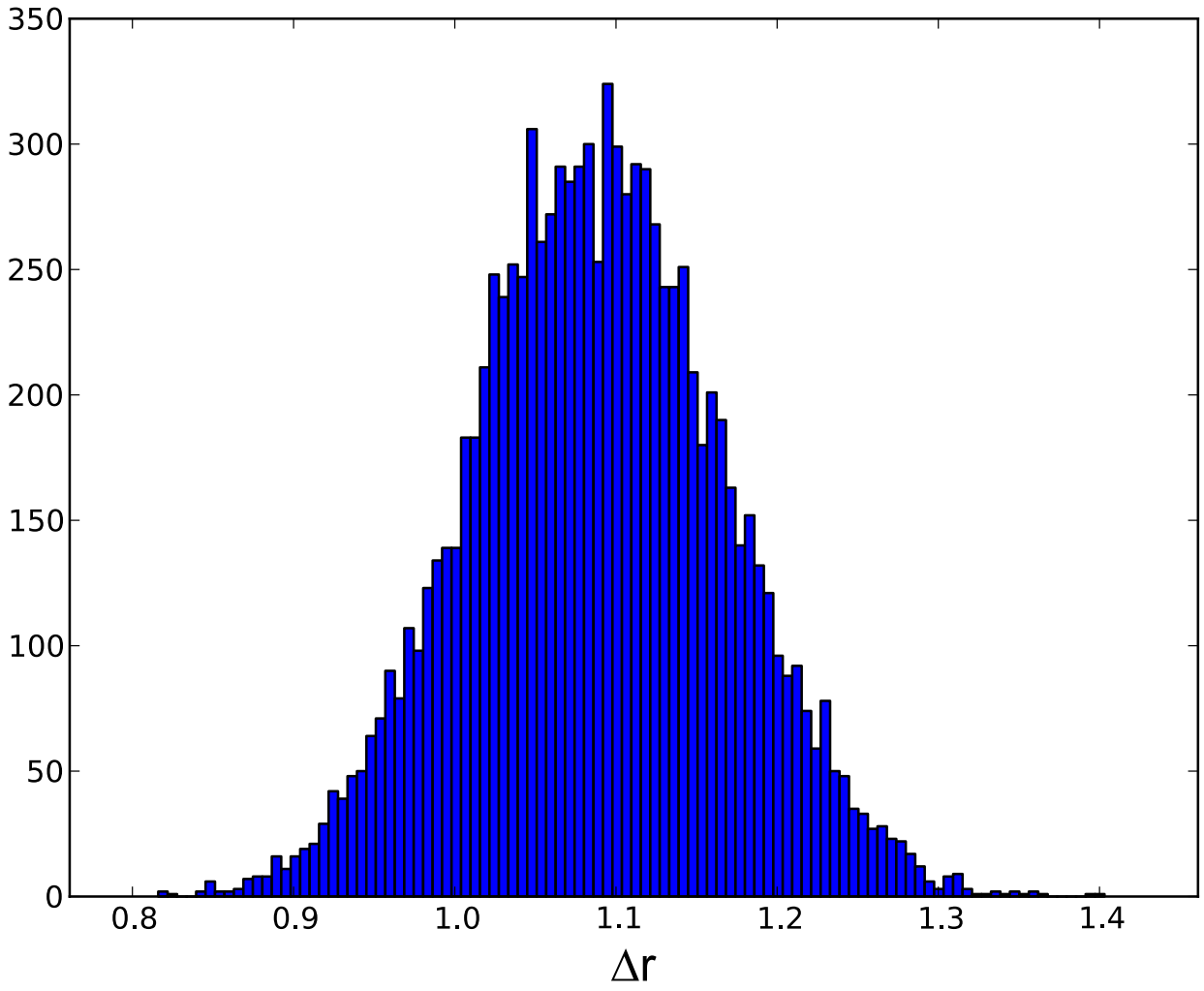


Рисунок 25 – Распределение прироста полноты системы семантико-синтаксического анализа по сравнению с системой, в которой синтаксический и семантический анализ выполняются раздельно

Доверительный интервал для F_1-меры при уровне значимости $\alpha = 0{,}05$ составил (0,4; 0,6) %. На рисунке 26 проиллюстрировано распределение прироста F_1-меры (ΔF_1) системы семантико-синтаксического анализа по сравнению с системой, в которой синтаксический и семантический анализ выполняются раздельно.

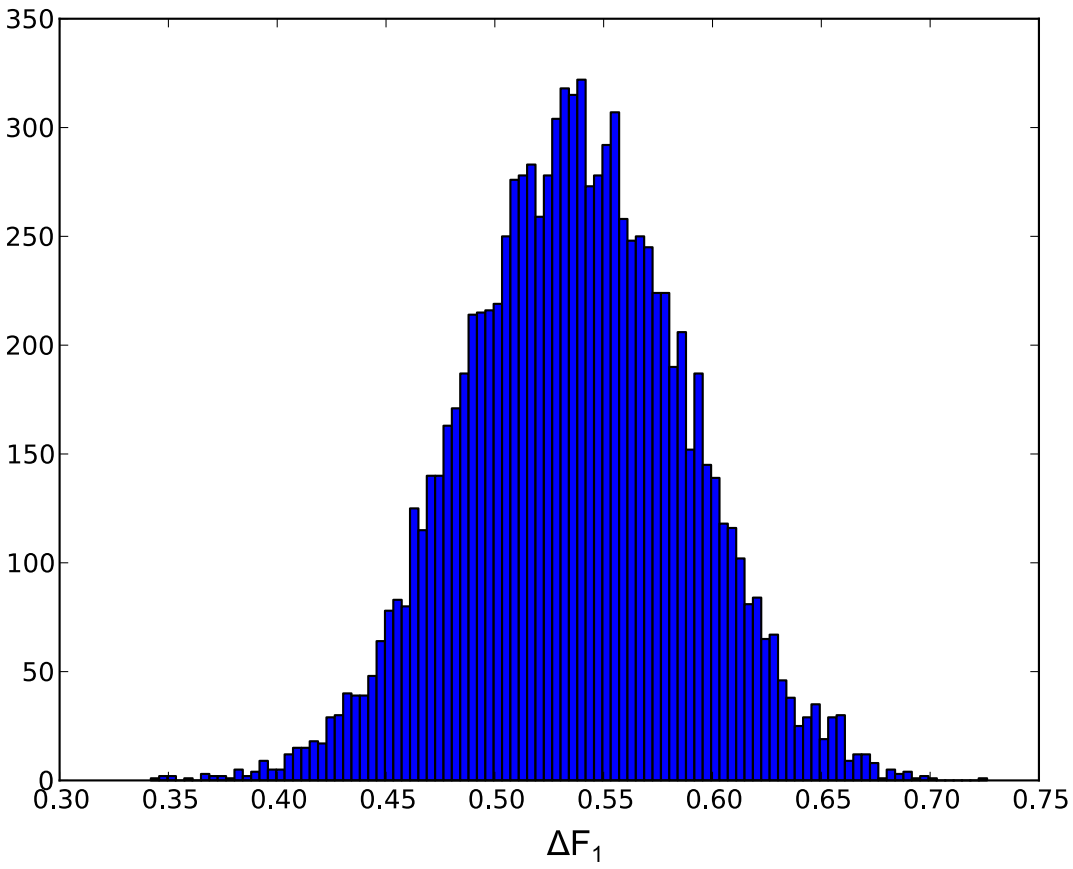


Рисунок 26 – Распределение прироста F_1-меры системы семантико-синтаксического анализа по сравнению с системой, в которой синтаксический и семантический анализ выполняются раздельно

Экспериментальное сравнение методов фильтрации исправлений показало, что в целом метод на основе ансамбля бинарных классификаторов работает лучше статистико-эвристического критерия. Он показывает большую точность при небольшом снижении полноты. Однако стоит отметить, что достаточно простой критерий также позволяет достаточно хорошо отфильтровывать неверные исправления. Тестирование показало, что точность внесения исправлений в синтаксическое дерево с помощью системы семантико-синтаксического составляет 82-85% в зависимости от метода проверки исправлений, параметров классификаторов, а также порога решающего правила. На рисунках 27 и 28, а также в приложении 4, представлены примеры, в которых

система семантико-синтаксического анализа исправила синтаксические структуры предложений. На этих рисунках красным цветом подсвечены синтаксические связи, которые были добавлены системой семантико-синтаксического анализа, в результате чего ей удалось также построить более полные ролевые структуры.

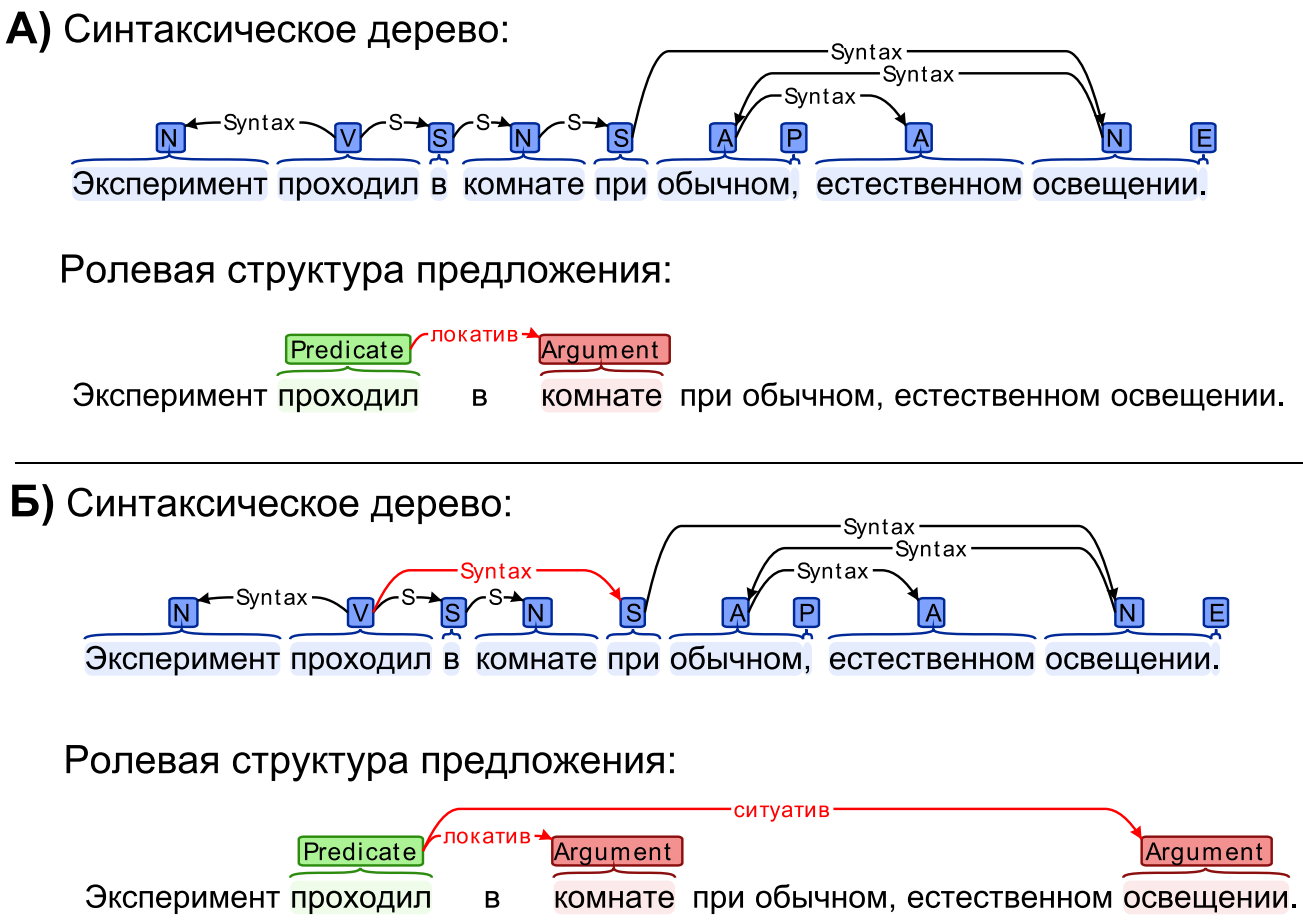


Рисунок 27 – Пример, в котором система семантико-синтаксического анализа исправила синтаксическую связь. На рисунке А) – пример работы MaltParser, на рисунке Б) – пример работы системы семантико-синтаксического анализа

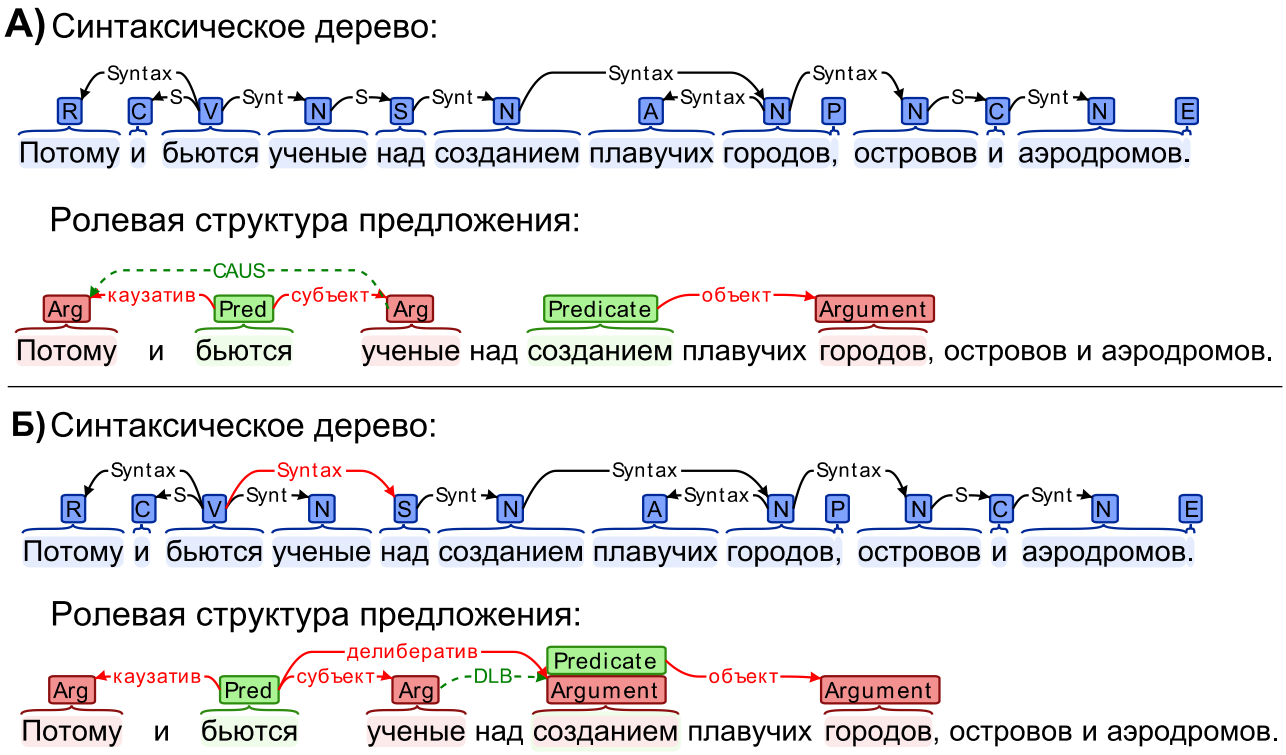


Рисунок 28 – Пример, в котором система семантико-синтаксического анализа исправила синтаксическую связь. На рисунке А) – пример работы MaltParser, на рисунке Б) – пример работы системы семантико-синтаксического анализа

Несмотря на то, что процедура корректировки системы семантико-синтаксического анализа работает с высокой точностью, в ряде случаев могут возникать ошибки, в результате которых правильная синтаксическая связь изменяется на неправильную. Во-первых, имеются случаи семантической омонимии, когда на уровне анализа одного предложения невозможно (даже для носителя языка) разрешить неоднозначность построения как семантической, так и синтаксической структуры предложения. Во-вторых, существуют ситуации, в которых два слова «конфликтуют» за право присоединить к себе один и тот же семантический аргумент, чтобы заполнить некоторую свою семантическую валентность. Если в семантическом словаре анализатора присутствуют знания о валентностях обоих слов, то такую ситуацию потенциально не сложно

разрешить. Однако в семантическом словаре присутствует информация о ролевых структурах далеко не всех предикатных слов. Хотя покрытие глаголов достаточно высокое, ролевые структуры многих отглагольных существительных остаются неизвестными. Поэтому для разрешения неоднозначности установления синтаксических связей для этих случаев были задействованы статистические методы. Но статистико-эвристический критерий и классификаторы на основе машинного обучения не всегда могут точно разрешить подобные ситуации.

На рисунке 29 представлен пример, в котором система семантико-синтаксического анализа правильно корректирует синтаксическое дерево, а на рисунке 30 представлен похожий пример, в котором корректировка дерева приводит к ошибке.

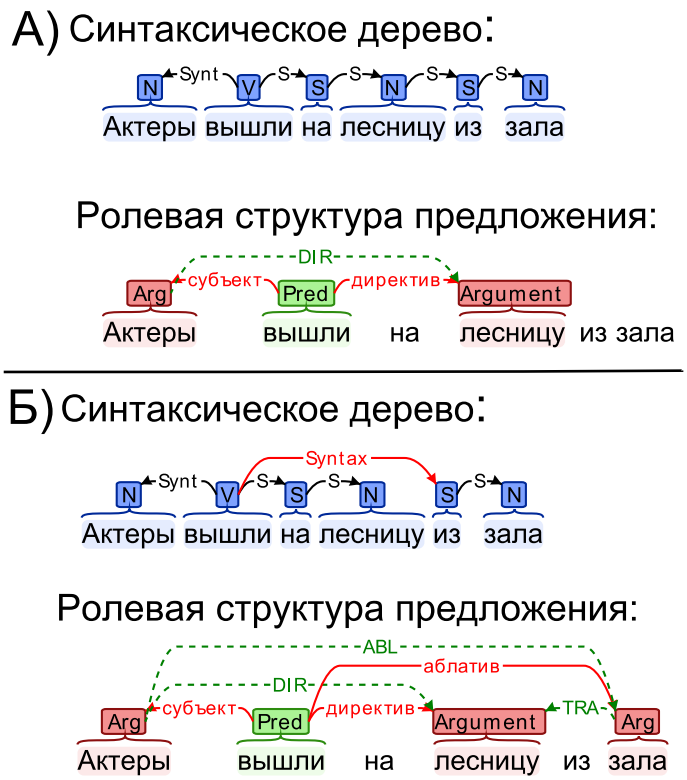


Рисунок 29 – Пример, в котором в результате корректировки синтаксического дерева была исправлена ошибка

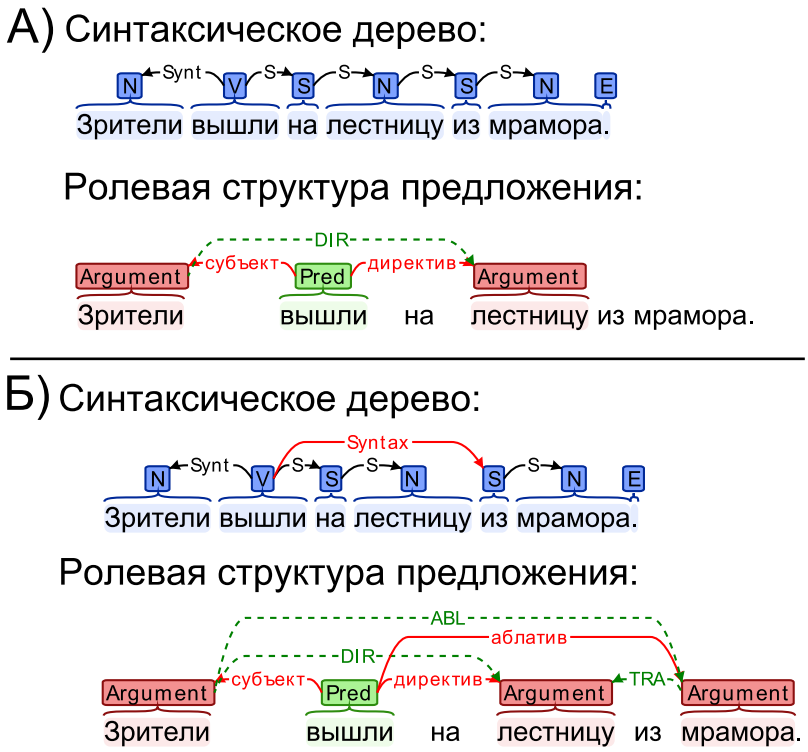


Рисунок 30 – Пример, в котором корректировка синтаксического дерева привела к ошибке

Повышение качества как синтаксического анализа, так и семантического анализа при использовании разработанного метода семантико-синтаксического анализа сильно связано с полнотой семантического словаря.

3.5.2. Оценка влияния метода семантико-синтаксического анализа на качество решения задачи определения ролевых структур высказываний

В проведенных экспериментах сравнивалось качество решения задачи определения ролевых структур высказываний системой семантико-синтаксического анализа, которую обозначим как «Сем.-син.», и системой, в которой синтаксический и семантический анализ выполняются раздельно, ее обозначим как «Базовая». В системе семантико-синтаксического анализа

использовался метод проверки исправлений с помощью ансамбля бинарных классификаторов на основе машинного обучения.

Тестирование системы семантико-синтаксического анализа проводилось аналогично тестированию семантического анализатора (эксперименты описаны в разделе 3.4): использовался семантически размеченный подкорпус (1 500 предложений, примерно 25 000 токенов без учета пунктуации, 3 300 ролей), рассчитывались точность p, полнота p и F_1-мера. Результаты представлены в таблице 6.

Таблица 6 – Сравнение качества семантического анализа системы семантико-синтаксического анализа с системой, в которой синтаксический и семантический анализ выполняются раздельно

Система	$\boldsymbol{p}$, %	$\boldsymbol{r}$, %	$\boldsymbol{F_1}$, %
Базовая	**89,6**	61,0	72,6
Сем.-син.	**89,6**	**62,7**	**73,8**

Результаты показывают, что применение разработанного метода семантико-синтаксического анализа ведет к повышению качества решения задачи определения ролевых структур высказываний. В частности, система семантико-синтаксического анализа имеет большую полноту – на 1,7%, что ведет к повышению F_1-меры на 1,2%. При этом точность не уменьшается.

За счет корректировки синтаксической структуры повышается доля правильно выявленных семантических аргументов, что в итоге позволяет поднять качество семантического анализа в целом. На рисунках 27 и 28 представлены примеры, в которых система семантико-синтаксического анализа построила более полные ролевые структуры по сравнению с системой, в которой синтаксический и семантический анализ выполняются раздельно.

3.6. Выводы

Представлены результаты экспериментальных исследований разработанных методов семантического и семантико-синтаксического анализа на размеченных корпусах русскоязычных текстов.

Получены результаты оценки качества решения задачи определения ролевых структур высказываний в текстах на русском языке. Результаты показывают, что качество решения этой задачи с помощью разработанного метода соответствует уровню современных семантических анализаторов для других языков с малым количеством языковых ресурсов.

Исследовано влияние разработанного метода семантико-синтаксического анализа на качество построения синтаксических деревьев зависимостей и на качество решения задачи определения ролевых структур высказываний. Система семантико-синтаксического анализа сравнивалась с синтаксическим анализатором MaltParser, который зарекомендовал себя как один из лучших синтаксических анализаторов на основе машинного обучения для русскоязычных текстов. Эксперименты показали, что разработанный метод семантико-синтаксического анализа позволяет повысить полноту установления синтаксических связей. Системе семантико-синтаксического анализа удается исправить наиболее сложные случаи установления синтаксических связей между предикатными словами и аргументами. Несмотря на то, что исходный синтаксический анализатор показывает высокую полноту – чуть менее 90%, с помощью системы семантико-синтаксического анализа удается восстановить еще более 10% отсутствующих синтаксических связей между предикатными словами и аргументами. Таким образом, впервые было исследовано влияние информации, заложенной в ролевой структуре высказываний, на качество установления синтаксических связей между предикатными словами и их

семантическими аргументами, и показана значимость этой информации для корректировки синтаксического дерева зависимостей.

Результаты экспериментов демонстрируют, что метод семантико-синтаксического анализа позволяет повысить также полноту и F_1-меру определения ролевых структур высказываний. Далее в главе 3 показывается важность применения семантического анализа при решении прикладных задач обработки текстов на естественном языке, а также демонстрируется эффект от прироста качества синтаксического и семантического анализа, достигаемый за счет применения разработанного метода семантико-синтаксического анализа.

ГЛАВА 4. ПРИМЕНЕНИЕ МЕТОДОВ СЕМАНТИЧЕСКОГО И СЕМАНТИКО-СИНТАКСИЧЕСКОГО АНАЛИЗА К ЗАДАЧАМ ОБРАБОТКИ ТЕКСТОВ НА ЕСТЕСТВЕННОМ ЯЗЫКЕ

В этой главе рассматриваются и оцениваются методы решения прикладных задач обработки текстов на естественном языке, в которых используются результаты семантического и семантико-синтаксического анализа.

4.1. Применение семантического и семантико-синтаксического анализа для построения семантической сети реляционно-ситуационной модели

Модель семантики текста на основе ролей является весьма эффективным инструментом при решении многих прикладных задач автоматической обработки текстов на естественном языке. Однако, как уже было отмечено ранее, она является довольно поверхностной, многие смысловые аспекты, выраженные в тексте, остаются за ее рамками. Более глубокое представление семантики текста отражено в реляционно-ситуационной модели текста. В ней синтаксемы и связи между ними образуют неоднородную семантическую сеть. Реляционно-ситуационная модель помимо информации о семантических ролях также учитывает значения свободных синтаксем и семантические отношения (СО). Разработанные автором диссертации методы определения ролевых структур высказываний описаны в главе 2, метод установления значений свободных синтаксем описан в работе [125]. В этом разделе рассмотрим метод установления семантических отношений, который необходим для построения полноценной реляционно-ситуационной модели.

Семантические отношения устанавливаются между аргументами предикатных слов. В методе используется словарь семантических отношений, в котором предикатным словам из семантического словаря сопоставлено множество СО [74]. Каждое СО характеризуется набором пар семантических ролей. Эти пары определяют какие роли должны иметь аргументы, чтобы участвовать в заданном СО. В таблице 7 представлен пример статьи в словаре семантических отношений для ситуации, описываемой предикатным словом «вызывать» в примере «ошибка врачей вызвала эпидемию». В работе использовался словарь семантических отношений, в котором задано 35 уникальных СО для около трех тысяч предикатных слов.

Таблица 7 – Пример статьи в словаре семантических отношений для предикатного слова «вызывать»

СО	**Роль главного аргумента СО**	**Роль зависимого аргумента СО**
CAUS	Каузатив	результатив
SIT	Ситуатив	каузатив

Для каждого предложения текста семантические отношения определяются отдельно. Пусть имеется ПА-структура предложения, построенная в результате семантического или семантико-синтаксического анализа. Каждое предикатное слово и его аргументы рассматриваются отдельно. Из словаря семантических отношений извлекается множество отношений, соответствующих ситуации, обозначенной заданным предикатным словом. Для каждого отношения ищется пара аргументов, которым назначены роли, соответствующие ролям в словарной статье заданной ситуации. Если пара найдена, то СО добавляется в семантическую сеть предложения. Алгоритм построения семантической сети представлен на рисунке 31.

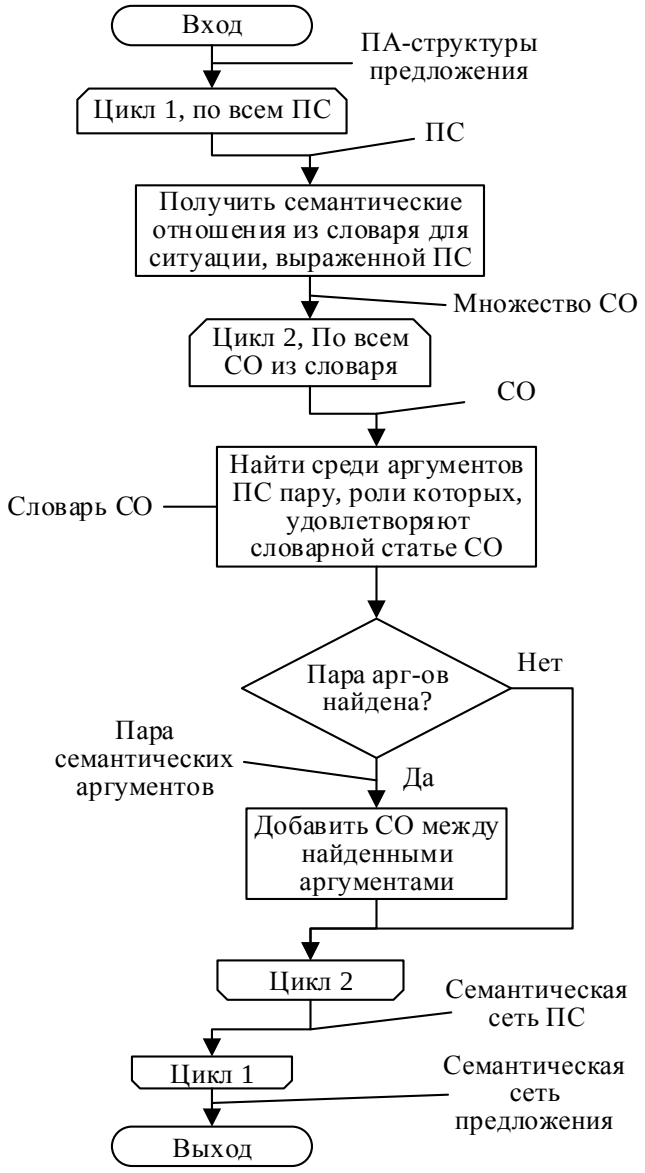


Рисунок 31 – Алгоритм построения семантической сети предолжения

Семантические отношения в построенной таким образом семантической сети позволяют находить и нормализовывать отношения между концептами предметной области, имеющими отражение в тексте. На рисунке 32 представлены автоматически построенные ролевые структуры двух предложений и выделенные в них семантические отношения (зеленые пунктирные стрелки, надпись на стрелке обозначает тип отношения.). В этих примерах несмотря на то, что аргументы «эпидемия» и «ошибка» имеют разные

семантические роли, а сами аргументы относятся к разным предикатным словам (соответственно «предикат» и «каузатив» на рисунке А), при предикатном слове «произойти»; соответственно «результатив» и «каузатив» при предикатном слове «привести» на рисунке Б)), между ними установлено одно и то же семантическое отношение «CAUS», обозначающее причинно-следственную связь.

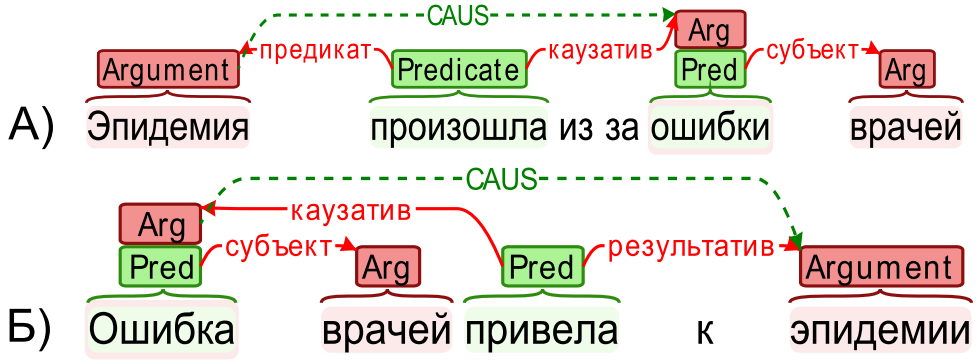


Рисунок 32 – Пример, в котором аргументы имеют разные семантические роли, но между ними установлено одно и то же семантическое отношение

Семантическая сеть может применяться для решения многих практических задач в области информационного поиска и извлечения информации. Далее в разделе 4.2 представлен метод, в котором она используется для вопросно-ответного поиска.

4.2. Применение семантического и семантико-синтаксического анализа для вопросно-ответного поиска

Задача вопросно-ответного поиска заключается в том, чтобы выдать пользователю ответ на вопрос, заданный на естественном языке. Существует множество формулировок этой задачи и типов вопросно-ответных систем. Ответ может искаться как в структурированных базах знаний, так и в коллекциях

неструктурированных текстовых документов. В этом разделе рассматривается задача вопросно-ответного поиска в метапоисковой системе.

У многих поисковых систем в Интернет модуль вопросно-ответного поиска либо отсутствует, либо он обладает ограниченными возможностями (например, отвечает только на вопросы «Где?», «Когда?»), поэтому ответы на вопросы пользователей, заданные на естественном языке, часто оказываются в глубине поисковой выдачи. Метапоисковая система делает запросы сразу к нескольким поисковым системам, получает от каждой из них сниппеты (фрагменты текста, которые выводятся рядом с ссылкой в поисковой выдаче) из топа ответов некоторой глубины, агрегирует все сниппеты в единую выдачу и ранижирует ее так, чтобы непосредственные ответы на вопросы, ссылки на релевантные документы и другая наиболее релевантная информация были в начале поисковой выдачи (рисунок 33).

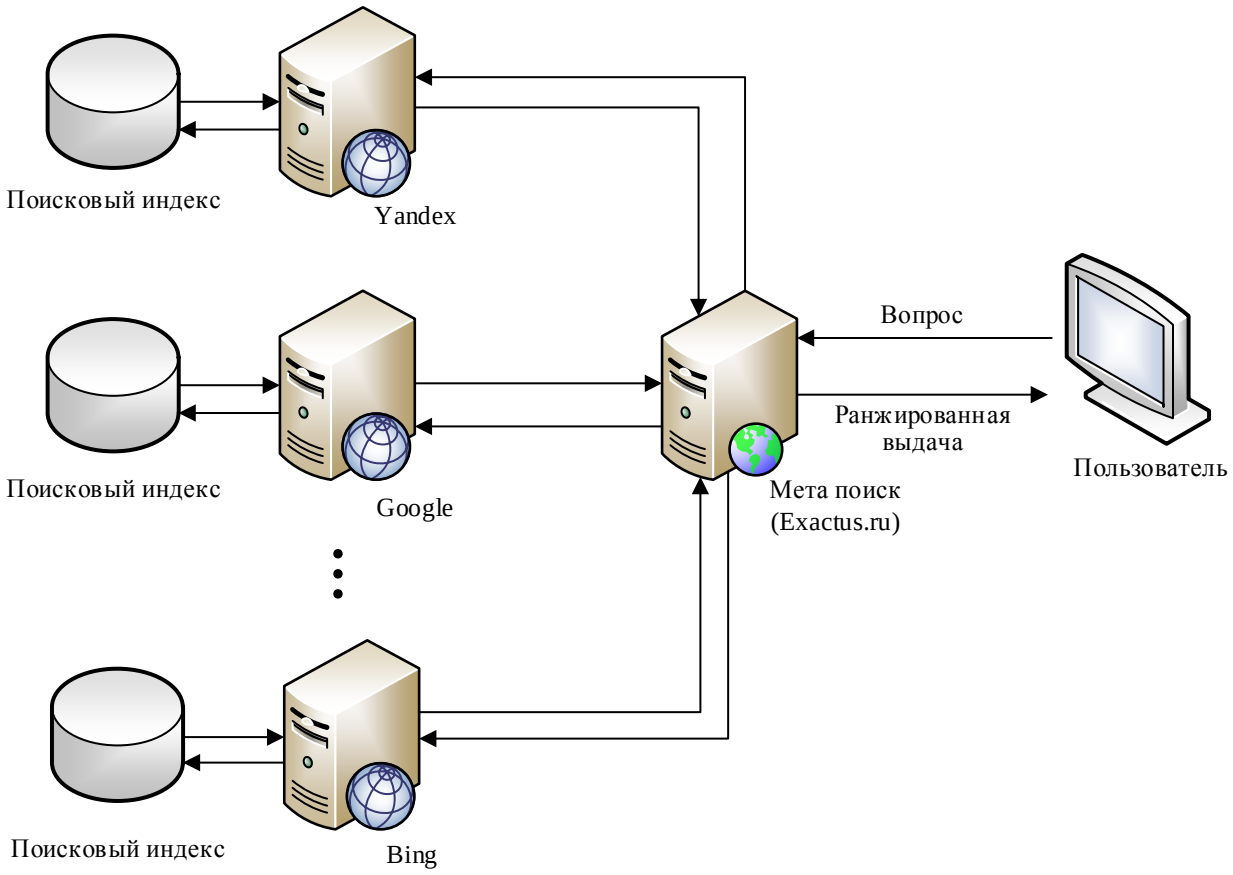


Рисунок 33 – Архитектура метапоисковой системы

В этом разделе предлагается метод ранжирования сниппетов для вопросно-ответного поиска в метапоисковой системе, в котором используется информация, полученная в результате реляционно-ситуационного анализа, который включает в себя семантико-синтаксический анализ, а также построение семантической сети (см. раздел 4.1). Метод позволяет находить ответы на категориальные вопросы (т.е. вопросы, в которых в начале присутствуют вопросительные слова: «кто», «что», «где», «когда» и др.), заданные на естественном языке. Описываются его экспериментальные исследования.

4.2.1. Метод ранжирования сниппетов для вопросно-ответного поиска в метапоисковой системе

Для ранжирования сниппетов, полученных от разных поисковых систем, для каждого из них вычисляется оценка релевантности r^{f_j} сниппета f_j (fragment) вопросительному запросу. В ходе вычисления этой оценки сравниваются лексика, семантические роли и семантические отношения запроса и сниппета.

Общий алгоритм определения оценки релевантности сниппетов запросу

Предполагается, что запрос состоит из одного предложения, поскольку запросы пользователей к поисковой системе в веб, как правило, являются краткими. Алгоритм легко может быть обобщен на случай, когда запрос состоит из несколько предложений.

Сниппет разбивается на предложения и для каждого предложения вычисляется его оценка релевантности предложению вопросительного запроса. Эта оценка учитывает как лексику, так и семантику предложений (семантические роли и отношения), поэтому назовем эту оценку лексико-семантической (ЛС-оценка). Наряду с ЛС-оценкой вычисляется лексическая оценка (Л-оценка) предложения, которая учитывает только совпадение лексики предложений сниппета и запроса. Она необходима в дальнейшем для оценки релевантности всего сниппета запросу по лексике. Отметим, что при сопоставлении лексики, стоп-слова, а также вопросительные слова запроса такие как «что», «где» и др. не учитываются. Алгоритм вычисления Л- и ЛС- оценок предложения сниппета представлен в следующем пункте.

Кроме этого, определяется оценка соответствия лексики сниппета запросу по всем предложениям – Л-оценка сниппета (r_l^f). Она вычисляется как отношение мощности пересечения множества лемм слов сниппета (L_f) и множества лемм слов запроса (L_q) к мощности множества лемм слов запроса:

$$r_l^f = \frac{|L_f \cap L_q|}{|L_q|}. \quad (8)$$

Итоговая ЛС-оценка сниппета представляет собой линейную свертку максимальной ЛС-оценки предложения сниппета и Л-оценки сниппета:

$$r_{ls}^f = \alpha_{ls} * \max_{s_i}(\mathrm{r}_{\mathrm{ls}}^{\mathrm{s_i}}) + \alpha_l * r_l^f. \quad (9)$$

Добавление r_l^f в формулу релевантности призвано повысить релевантность тех сниппетов, в которых присутствует больше слов из запроса вне зависимости от того, как они распределены по предложениям, при прочих равных оценках. На практике коэффициент при лексико-семантической оценке должен быть значительно больше чем коэффициент при лексической оценке $\alpha_{ls} > \alpha_l$, Например, в экспериментах использовались эмпирически подобранные коэффициенты $\alpha_{ls} = 0{,}9;\ \alpha_l = 0{,}1$.

Для каждого сниппета вычисляется также полная Л-оценка r_{fl}^f, которая представляет сумму всех Л-оценок предложений сниппета:

$$r_{fl}^f = \sum_i r_l^{s_i}. \quad (10)$$

После того как вычислены полные Л-оценки и ЛС-оценки всех сниппетов рассчитывается финальная оценка релевантности каждого сниппета r^f. Она представляет собой линейную свертку ЛС-оценки сниппета и полной Л-оценки сниппета нормированной на максимальную полную-оценку среди всех сниппетов:

$$r^f = \beta_{ls} * r_{ls}^f + \beta_l * \frac{r_l^f}{\max_j r_l^{f_j}}. \quad (11)$$

На практике нормированная полная Л-оценка сниппета должна мало влиять на релевантность сниппета вопросительному запросу. Она добавляется в формулу релевантности для того, чтобы при прочих равных оценках повысить релевантность сниппетов с большим количеством лексики, совпавшей с лексикой запроса, и большим размером. Более крупные сниппеты с большей вероятностью содержат ответ нежели чем короткие сниппеты, что может помочь при поиске, если оценки, опирающиеся на семантику, малы. Кроме того, крупные сниппеты, как правило, содержат более развернутый ответ. В экспериментах использовались эмпирически подобранные коэффициенты $\beta_{ls} = 0{,}95$, $\beta_l = 0{,}05$.

На рисунке 34 изображен общий алгоритм расчета релевантности сниппетов вопросительному запросу.

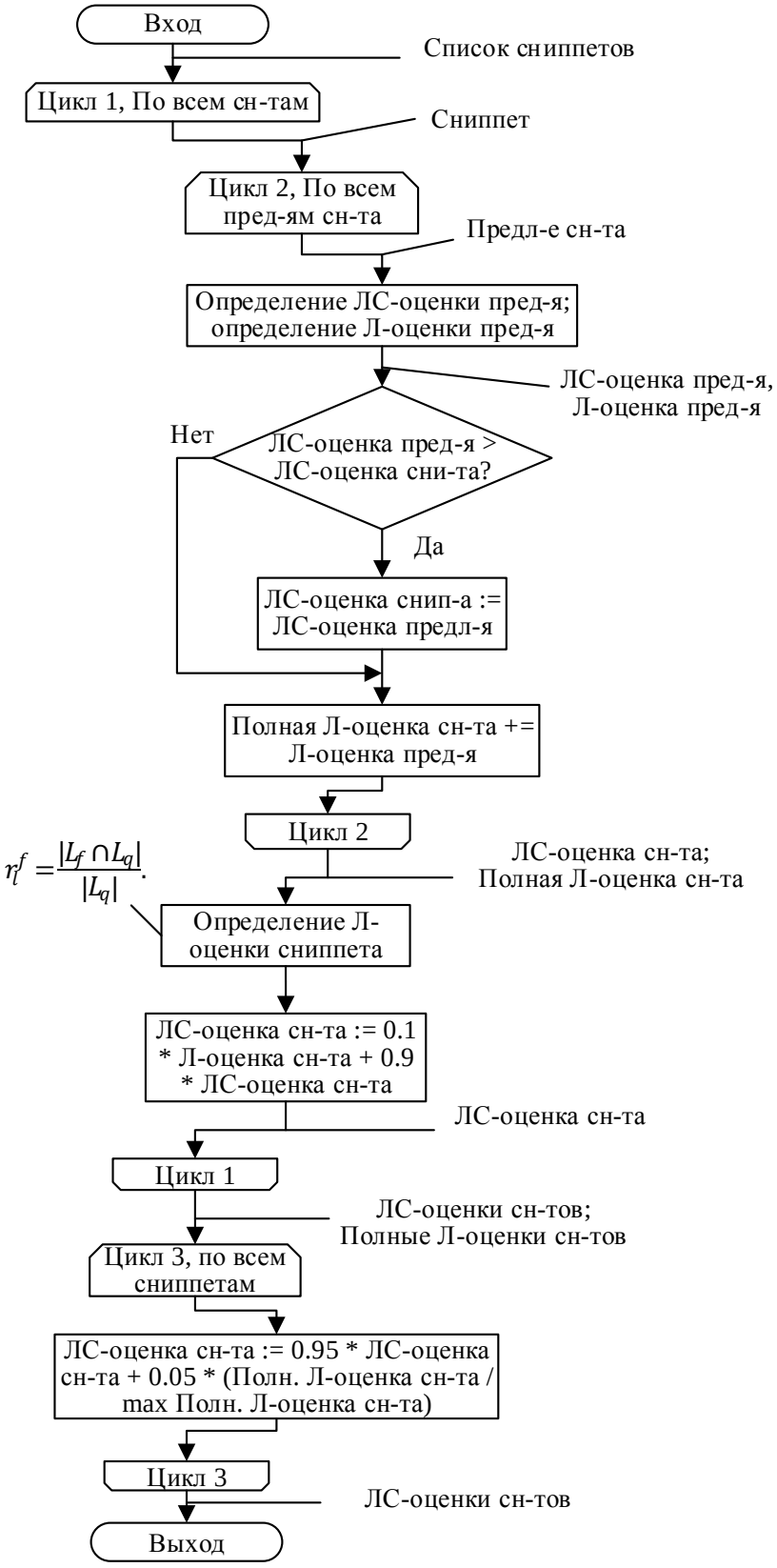


Рисунок 34 – Общий алгоритм расчета релевантности сниппетов вопросительному запросу

Алгоритм определения лексико-семантической оценки релевантности предложения сниппета запросу

Лексико-семантическая (ЛС) $r_{ls}^{s} \in [0,1]$ оценка релевантности предложения сниппета вопросительному запросу вычисляется как линейная свертка трех компонент:

- лексическая оценка (Л-оценка) предложения r_{l}^{s} (lexis) – оценивает близость запроса и предложения сниппета по лексике;
- оценка семантических ролей (СР-оценка) r_{sr}^{s} (semantic roles) – оценивает близость запроса и сниппета по предикатно-аргументным структурам, используя семантические роли аргументов;
- оценка семантических отношений (СО-оценка) r_{sn}^{s} (semantic net) – оценивает близость запроса и сниппета по семантическим отношениям семантической сети.

$$r_{ls}^{s} = \gamma_{l} * r_{l}^{s} + \gamma_{sr} * r_{sr}^{s} + \gamma_{sn} * r_{sn}^{s} \quad (12)$$

Здесь $\gamma_{l} \in [0,1]$, $\gamma_{sr} \in [0,1]$, $\gamma_{sn} \in [0,1]$ – параметры алгоритма такие, что $\gamma_{l} + \gamma_{sr} + \gamma_{sn} = 1$.

Лексическая оценка релевантности вычисляется как отношение мощности пересечения множества лемм запроса L_{q} с множеством лемм предложения сниппета L_{s} к мощности множества лемм запроса. При этом эти множества не включают леммы стоп-слов: предлогов, союзов, пунктуации и др., кроме того в эти множества не входят вопросительные слова запроса. Если предложение сниппета является вопросительным, то релевантность занижается, путем умножения Л-оценки на понижающий коэффициент $\delta_{l} \in (0,1)$, который является параметром алгоритма. Это необходимо, чтобы уменьшить значимость вопросительных предложений по сравнению с утвердительными, поскольку в сниппетах поисковиков часто вместо ответа на вопрос находится сам вопрос.

Кроме того, ответ в виде утвердительного предложения является более предпочтительным для пользователя. На практике релевантность вопросительных предложений необходимо занижать примерно в два раза.

$$r_l^s = \delta_l \frac{|L_q \cap L_s|}{|L_q|}, \delta_l = \begin{cases} 1, \text{если s – утвердительное предл.} \\ \delta_l^q \in (0,1), \text{если s – вопросительное предл.} \end{cases} \quad (13)$$

Рассмотрим метод расчета СР-оценки. Пусть в запросе определено P_q ПА-структур. СР-оценка релевантности предложения сниппета запросу r_{sr}^s представляет собой взвешенную нормализованную сумму оценок близости отдельных ПА-структур запроса и предложения сниппета $r_{sr}^{p_i}$ $(i = 1, \dots P_q)$. Пусть в предложении сниппета присутствует P_s ПА-структур. Каждая структура запроса p_i $(i = 1, \dots P_q)$ сопоставляется со всеми ПА-структурами предложения сниппета p_j $(j = 1, \dots P_s)$ и для каждой такой пары вычисляется оценка сходства $r_{sr}^p(p_i, p_j)$. Для заданной ПА-структуры p_i запроса выбирается структура предложения сниппета с максимальной оценкой сходства, и эта оценка становится оценкой $r_{sr}^{p_i}$:

$$r_{sr}^{p_i} = \max_j r_{sr}^p(p_i, p_j). \quad (14)$$

Каждая оценка сходства пары ПА-структур $r_{sr}^p(p_i, p_j)$ в свою очередь является суммой оценок сходства семантических аргументов $r_{sr}^{pa}(a^q, a^s)$ в ПА-структуре запроса и структуре предложения сниппета. Пусть в ПА-структуре запроса K аргументов: $p^q = \{a_1^q, a_2^q, \dots, a_K^q\}$, а в ПА-структуре предложения сниппета M аргументов $p^s = \{a_1^s, a_2^s, \dots, a_M^s\}$. Каждый семантический аргумент из ПА-структуры запроса, которому назначена семантическая роль $a_k^q \in p^q$, $(k = \overline{1, K})$ сопоставляется с каждым семантическим аргументов ПА-структуры

предложения сниппета $a_l^s \in p^s, (m = \overline{1,M})$, в результате чего вычисляется оценка сходства аргументов $r_{sr}^{pa}(a_k^q, a_m^s)$, $(k = \overline{1,K}, m = \overline{1,M})$.

Представим семантический аргумент в виде четверки:

$$a =< role, pred, syn, neg >. \quad (15)$$

Где $role$ – семантическая роль; $pred$ – лемма предикатного слова аргумента или идентификатор словарной статьи предикатного слова; syn – синтаксическое поддерево аргумента; neg – флаг, указывающий на наличие в тексте признаков отрицания аргумента, например, в отрывке «*бумагу не изготавливают из нефти*» аргумент «бумагу» имеет признаки отрицания.

Пусть имеется семантический аргумент запроса $a^q =< role^q, pred^q, syn^q, neg^q >$ и семантический аргумент сниппета $a^s =< role^s, pred^s, syn^s, neg^s >$. Рассмотрим алгоритм вычисления оценки сходства семантических аргументов $r_{sr}^{pa}(a^q, a^s)$.

1) Зададим начальное значение $r_{sr}^{pa}(a^q, a^s) = 0$.
2) Если $role^s \neq role^q$, то оценка $r_{sr}^{pa}(a^q, a^s) = 0$, выход. Иначе перейти к следующему шагу.
3) Если синтаксической вершиной syn^s является вопросительное слово, то оценка $r_{sr}^{pa}(a^q, a^s) = 0$, выход. Иначе перейти к следящему шагу.
4) Проверить совпадение по предикатному слову. Если $pred^s = pred^q$ (совпадают леммы предикатных слов или идентификаторы словарных статей предикатных слов), то модифицировать $r_{sr}^{pa}(a^q, a^s) := r_{sr}^{pa}(a^q, a^s) + w_p$. Вес $w_p > 0$. Перейти к следующему шагу.
5) Если синтаксической вершиной syn^q, является вопросительное слово, то перейти к шагу 8). Иначе перейти к следующему шагу.

6) Сравнить синтаксические поддеревья аргумента запроса syn^q и аргумента сниппета syn^s. Оценка сходства синтаксических поддеревьев $sim(syn^s, syn^q)$ вычисляется путем сопоставления лемм слов с учетом уровня слов в поддереве аргумента запроса и аргумента сниппета. Модифицировать $r_{sr}^{pa}(a^q, a^s) := r_{sr}^{pa}(a^q, a^s) + w_a * sim(syn^s, syn^q)$. Вес $w_a > 0$. Перейти к следующему шагу.
7) Если есть совпадение по предикатным словам и совпадение по синтаксическим поддеревьям, то модифицировать $r_{sr}^{pa}(a^q, a^s) := r_{sr}^{pa}(a^q, a^s) + w_f$. Вес $w_f > 0$. Выход.
8) Если нет совпадения по предикатным словам, то модифицировать $r_{sr}^{pa}(a^q, a^s) := r_{sr}^{pa}(a^q, a^s) + w_{q1}$, вес $w_{q1} > 0$, выход. Иначе перейти к следующему шагу.
9) Если $neg^s \neq neg^q$, выход. Иначе перейти к следующему шагу.
10) Модифицировать $r_{sr}^{pa}(a^q, a^s) := r_{sr}^{pa}(a^q, a^s) + w_{q2}$, вес $w_{q2} > 0$ может быть модифицирован набором эвристик, например, если аргумент сниппета представлен местоимением, то вес w_{q2} снижается. На этом шаге также выделяется ответ – синтаксическое поддерево аргумента сниппета заданной максимальной глубины. Выход.
11) Конец.

Величины $w_p, w_a, w_f, w_{q1}, w_{q2}$ являются параметрами алгоритма, их можно варьировать в соответствии с эвристическими предположениями о значимости вклада компонентов в оценку релевантности или, например, настраивать с помощью методов оптимизации на заданной размеченной выборке.

Среди всех оценок пар сходства семантических аргументов для заданного аргумента запроса и заданной ПА-структуры предложения сниппета находится максимальная, которая в конечном счете участвует в расчете оценки релевантности всей ПА-структуры:

$$r_{sr}^{p}(p_i, p_j) = \sum_{k=1}^{K_i} \max_{m} r_{sr}^{ra}(a_k^q, a_m^s). \quad (16)$$

Оценка сходства между ПА-структурами $r_{sr}^{p}(p_i, p_j)$ может быть больше единицы. При этом максимальная оценка для разных по размеру и признакам ПА-структур может отличаться. Структуры, в которых присутствует больше семантических аргументов, которым назначены роли, потенциально могут получить большую оценку, чем те, в которых семантических аргументов меньше. Это соответствует представлению о более высоком вкладе в оценку релевантности запроса сниппету наличия сходства более крупных ПА-структур по сравнению с наличием сходства мелких ПА-структур. Кроме того, структуры запроса, в которых присутствует аргумент-вопросительное слово также считаются более важными при оценке релевантности по семантическим аргументам.

В итоге СР-оценка запроса и предложения сниппета определяется следующим образом:

$$r_{sr} = \delta_{sr} \frac{\sum_{i=1}^{P_q} \max_{j} r_{sr}^{p}(p_i, p_j)}{Z_{sr}}. \quad (17)$$

Здесь Z_{sr} – коэффициент нормализации, который необходим, чтобы конечная оценка релевантности по семантическим ролям не выходила за рамки отрезка [0,1]. Z_{sr} вычисляется как сумма всех максимально возможных оценок сходства ПА-структур запроса вне зависимости от ПА-структур предложения сниппета. Коэффициент $\delta_{sr} \in [0,1)$ понижает СР-релевантность предложения в соответствии с набором эвристик. Эти эвристики учитывают различную лингвистическую информацию о предложении, например, тип предложения:

утвердительное, предложение-условие («Если бы произошел финансовый кризис в 2012 году…»), вопрос. Релевантность утвердительного предложения должна быть выше, чем предложения-условия, или вопросительного предложения.

СО-оценка r_{sn}^{s} рассчитывается по схожему принципу, однако семантические отношения не привязаны к предикатным словам и ПА-структурам, поэтому СО-оценка вычисляется как нормализованная сумма оценок сходства семантического отношения в запросе с семантическим отношением в предложении сниппета $r_{sn}^{sr}(r^q, r^s)$.

Пусть в предложении запроса имеется R^q отношений, в предложении сниппета – R^s отношений. Представим отношение в виде тройки:

$$r =< type, syn_{left}, syn_{right} >. \quad (18)$$

Здесь $type$ – тип семантического отношения; syn_{left} – синтаксическое поддерево, в вершине которого находится слово, из которого выходит семантическое отношение; syn_{right} – синтаксическое поддерево, в вершине которого находится слово, в которое ведет семантическая отношение. Каждое отношение запроса $r^q =< type^q, syn_{left}^q, syn_{right}^q >$ попарно сравнивается с отношением предложения сниппета $r^s =< type^s, syn_{left}^s, syn_{right}^s >$. Рассмотрим алгоритм сравнения отношений.

1) Зададим начальное значение $r_{sn}^{sr}(r^q, r^s) = 0$.
2) Если $type^q \neq type^s$, то оценка $r_{sn}^{sr}(r^q, r^s) = 0$, выход. Иначе перейти к следующему шагу.
3) Если синтаксической вершиной syn_{left}^s или syn_{right}^s является вопросительное слово, то оценка $r_{sn}^{sr}(r^q, r^s) = 0$. Иначе перейти к следящему шагу.

4) Если синтаксической вершиной syn_{left}^{q} или syn_{right}^{q} является вопросительное слово, то перейти к шагу 8). Иначе перейти к следующему шагу.
5) Оценить сходство синтаксических поддеревьев $sim\left(syn_{left}^{q}, syn_{left}^{s}\right) \in [0,1]$. Модифицировать $r_{sn}^{sr}(r^q, r^s) \coloneqq r_{sn}^{sr}(r^q, r^s) + w_{sl} * sim(syn_{left}^{q}, syn_{left}^{s})$. Вес $w_{sl} > 0$. Перейти к следующему шагу.
6) Оценить сходство синтаксических поддеревьев $sim\left(syn_{right}^{q}, syn_{right}^{s}\right) \in [0,1]$. Модифицировать $r_{sn}^{sr}(r^q, r^s) \coloneqq r_{sn}^{sr}(r^q, r^s) + w_{sr} * sim(syn_{right}^{q}, syn_{right}^{s})$. Вес $w_{sr} > 0$. Перейти к следующему шагу.
7) Модифицировать $r_{sn}^{sr}(r^q, r^s) \coloneqq \delta_{sn} * r_{sn}^{sr}(r^q, r^s),\ \delta_{sn} > 1$. Коэффициент δ_{sn} необходим для того, чтобы повысить различие между оценками сходства, при высоких суммарных оценках сходства синтаксических поддеревьев и при низких. Выход.
8) Пусть вопросительное слово является вершиной syn_{left}^{q}. Модифицировать $r_{sn}^{sr}(r^q, r^s) \coloneqq r_{sn}^{sr}(r^q, r^s) + w_{rq}$. Перейти к следующему шагу.
9) Оценить сходство синтаксических поддеревьев $sim\left(syn_{right}^{q}, syn_{right}^{s}\right) \in [0,1]$. Модифицировать $r_{sn}^{sr}(r^q, r^s) \coloneqq r_{sn}^{sr}(r^q, r^s) + w_{lq} * sim\left(syn_{right}^{q}, syn_{right}^{s}\right)$. Перейти к следующему шагу.
10) Модифицировать $r_{sn}^{sr}(r^q, r^s) \coloneqq \delta_{sn} * r_{sn}^{sr}(r^q, r^s)$. Выход.
11) Конец.

Среди всех оценок пар сходства семантических отношений для заданного семантического отношения запроса находится максимальная, которая в конечном счете участвует в расчете СО-оценки r_{sn}:

$$r_{sn} = \frac{\sum_{i=1}^{R_q} \max_j r_{sn}^{sr}\left(r_i^q, r_j^s\right)}{Z_{sn}}. \quad (19)$$

Здесь Z_{sn} коэффициент нормализации, который необходим, чтобы конечная оценка релевантности по семантическим отношениям не выходила за рамки отрезка [0;1]. Z_{sn} вычисляется как сумма всех максимально возможных оценок сходства семантических отношений предложения запроса вне зависимости от отношений предложения сниппета. Значения w_{sl}, w_{sr}, w_{rq}, w_{lq}, δ_{sn} – являются параметрами алгоритма.

4.2.2. Экспериментальное исследование метода ранжирования сниппетов для вопросно-ответного поиска в метапоисковой системе

Для экспериментальной проверки метода ранжирования сниппетов для вопросно-ответного поиска был создан размеченный корпус вопросительных запросов и ответов. Двое разметчиков сформировали список вопросительных запросов на русском языке. Вопросы являются категориальными и содержат вопросительные слова «кто» (и др. формы «кем», «кому» и т.д.), «что» (и др. формы «чем», «чему» и т.д.), «кто такой», «что такое», «зачем», «когда», «где», «сколько», «куда», «откуда», «почему». Всего было сформулировано 195 запросов. Они были заданы четырем поисковым системам в Интернет: Yandex, Google, Bing и Yahoo. От каждой системы был получен топ ответов глубиной 20. Каждый ответ состоит из поискового сниппета и его заголовка. Таким образом, для каждого заданного вопроса был получен список, состоящий из около 50-70 ответов без повторений: сниппетов и их заголовков.

Для каждого запроса разметчики оценили каждый ответ поисковой системы и определили, является ли он ответом на поставленный вопрос. Оценивался и сниппет, и его заголовок. Ответ поисковой системы считался ответом на поставленный вопрос если:

- сниппет или его заголовок содержат прямой ответ на вопрос;

  Пример:

  Вопрос: «Из чего делают сахар?»

  Ответ:

  Заголовок: «Из чего делают сахар ? - Большой ...»

  Сниппет: «Сахар (или сахароза) делают в основном из *сока сахарного тростника или сахарной свеклы*.»

- если информация в заголовке соотносится с заданным вопросом, а в сниппете содержится краткий правильный ответ;

  Пример:

  Вопрос: «Кто написал «что делать»?»

  Ответ:

  Заголовок: «Ответы@Mail.Ru: кто написал произведение что делать?»

  Сниппет: «*Николай Гаврилович Чернышевский*. Нравится Пожаловаться. 3 ответа. опорполп Мастер (2258) 4 года назад. Чернышевский. Нравится Пожаловаться.",»

- если в заголовке или сниппете отсутствует прямой подробный ответ на вопрос, но содержится корректная информация, которую человек интуитивно может интерпретировать как ответ.

  Пример:

  Вопрос: «Чем стричь газоны?»

  Ответ:

  Заголовок: «Стрижка газона - ее особенности»

  Сниппет: «Стрижка газона по видам. Для партерных газонов можно использовать готовый рулонный газон , который необходимо подстригать на высоту 5 см. — это в первые 2 года. *Газона хорошего у нас пока нет, еще только строимся,*

но траву стрижем . Самое первое, что подарили мне дети после покупки деревенского дома — газонокосилку.»

Качество автоматического ранжирования для вопросно-ответного поиска оценивалось по точности p, которая вычислялась как отношение количества вопросов, ответ на который система дала на глубине не больше чем $d > 0$, ко всему количеству запросов. Точность оценивалась при $d = 1$, $d = 2$, $d = 3$, $d = 4$. Наиболее важным показателем в вопросно-ответном поиске считается точность при $d = 1$ [169, 170] – это оценка того, как часто ответ на заданный вопрос находится в первом сниппете или заголовке выдачи метапоисковой системы. Например, это важно для задачи извлечения непосредственно ответа на вопрос.

Сравнивались четыре системы ранжирования.

- система, которая ранжирует ответы случайным образом (среднее по 5 прогонам). Обозначим ее как «Случ. ранж.».
- Система, которая ранжирует ответы на основе лишь лексического критерия – все веса и коэффициенты при семантических оценках – нули. Обозначим ее как «Лексич. ранж.».
- Система, которая использует описанный выше алгоритм вычисления релевантности с учетом семантической информации, полученной от системы, в которой синтаксический и семантический анализ выполняются раздельно. Обозначим ее как «Сем. ранж. Сем. ан.».
- Система, которая использует описанный выше алгоритм вычисления релевантности с учетом семантической информации, полученной от системы семантико-синтаксического анализа. Обозначим ее как «Сем. ранж. Сем.-син. ан.».

Результаты представлены в таблице 8 и на рисунке 35.

Таблица 8 – Точность ранжирования сниппетов для вопросно-ответного поиска

Максимальная глубина (d)	**Системы ранжирования**			
	Сем. ранж. Сем. ан.	Сем. ранж. Сем.-син. ан.	Лексич. ранж.	Случ. ранж.
1	53,9	**58,0**	46,1	26,0
2	73,1	**75,1**	64,8	40,1
3	79,3	**79,8**	71,5	52,8
4	82,9	**83,4**	75,6	62,3

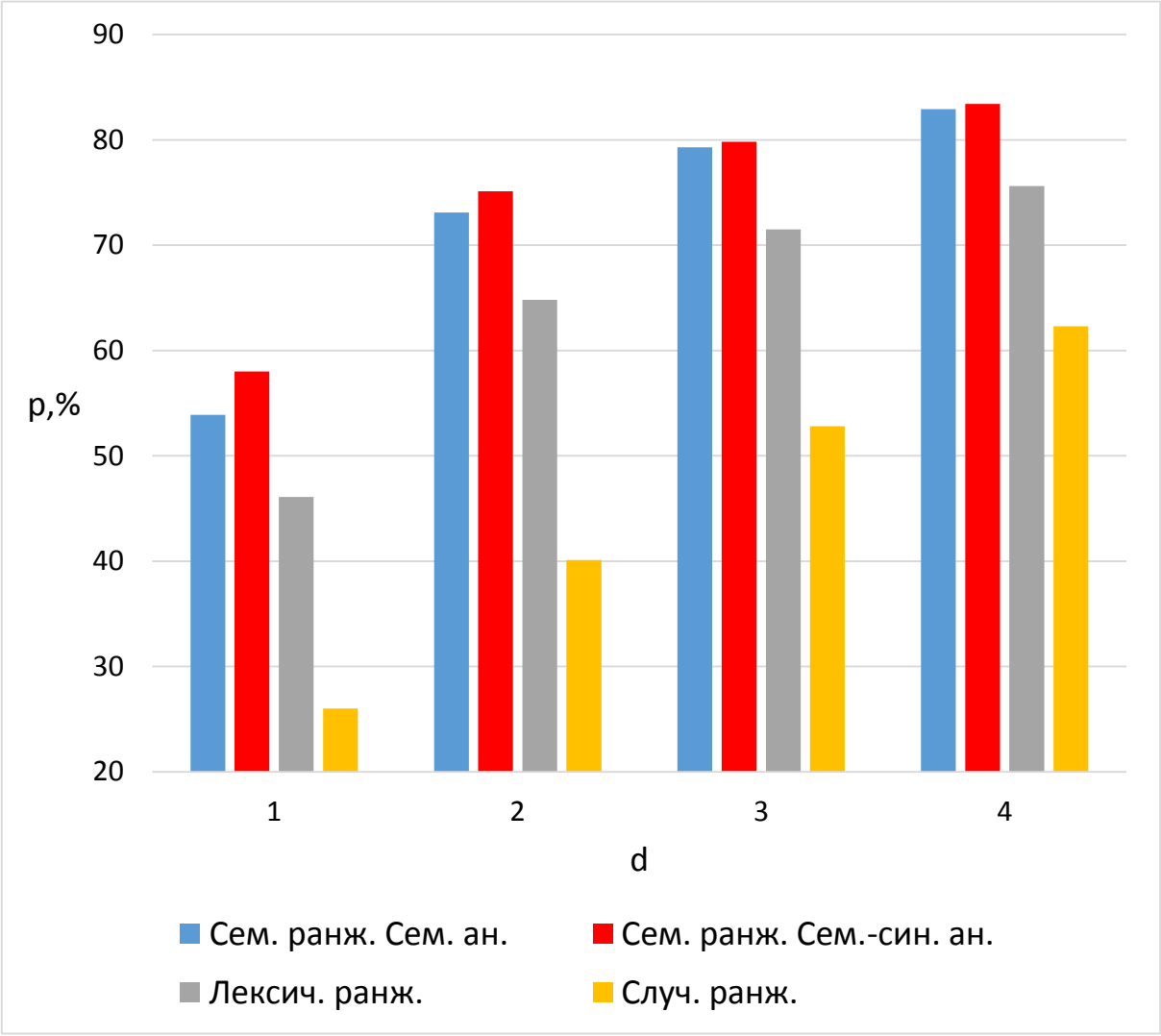


Рисунок 35 – Зависимость точности ранжирования $\boldsymbol{p}$ от максимальной глубины выдачи $\boldsymbol{d}$

Для оценки качества вопросно-ответного поиска часто используют также метрику mean reciprocal rank (MRR), которая вычисляется следующим образом:

$$MRR = \frac{1}{Q}\sum_{i=1}^{Q}\frac{1}{r_i}. \quad (20)$$

Здесь r_i – ранг первого правильного ответа на i-ый вопрос в выдаче; Q – общее количество вопросов. Эта метрика учитывает правильные ответы системы на небольшой глубине выдачи, которые не оказались первыми, но при этом штрафует их, снижая их вклад в итоговую оценку качества. Таким образом, она является более мягкой оценкой по отношению к точности при $d = 1$, но более жесткой по отношению к точности при $d > 1$.

Обычно максимальный ранг ограничен. В проведенных экспериментах использовалась метрика MRR, которая применялась на соревнования TREC по вопросно-ответному поиску, в ней максимальный ранг ограничивается 5 [171]. Сравнивались три способа ранжирования «Лексич. ранж.», «Сем. ранж. Сем. ан.», «Сем. ранж. Сем.-син. ан.». Результаты представлены в таблице 9.

Таблица 9 – Оценки MRR систем ранжирования сниппетов для вопросно-ответного поиска

Системы ранжирования	$\boldsymbol{MRR \cdot 100}$
Лексич. ранж.	59,0
Сем. ранж. Сем. ан.	67,2
Сем. ранж. Сем.-син. ан.	**69,6**

Результаты показывают следующее:

1) Разработанный критерий вносит большой вклад в качество ранжирования ответов поисковых систем: точность на 30-35% выше по сравнению со случайным ранжированием.

2) Учет семантической структуры предложения при оценке релевантности ответов в вопросно-ответном поиске позволяет повысить точность на 10-12% и значение MRR на более чем 8% по сравнению с лексическим критерием ранжирования, а также позволяет извлекать из сниппета сам ответ на вопрос. В качестве иллюстрации на рисунке 36 представлен скриншот веб-интерфейса модуля метапоиска системы Exactus.ru, в котором реализован описанный метод. В этом примере ответы на вопросы в сниппетах выделены курсивом.

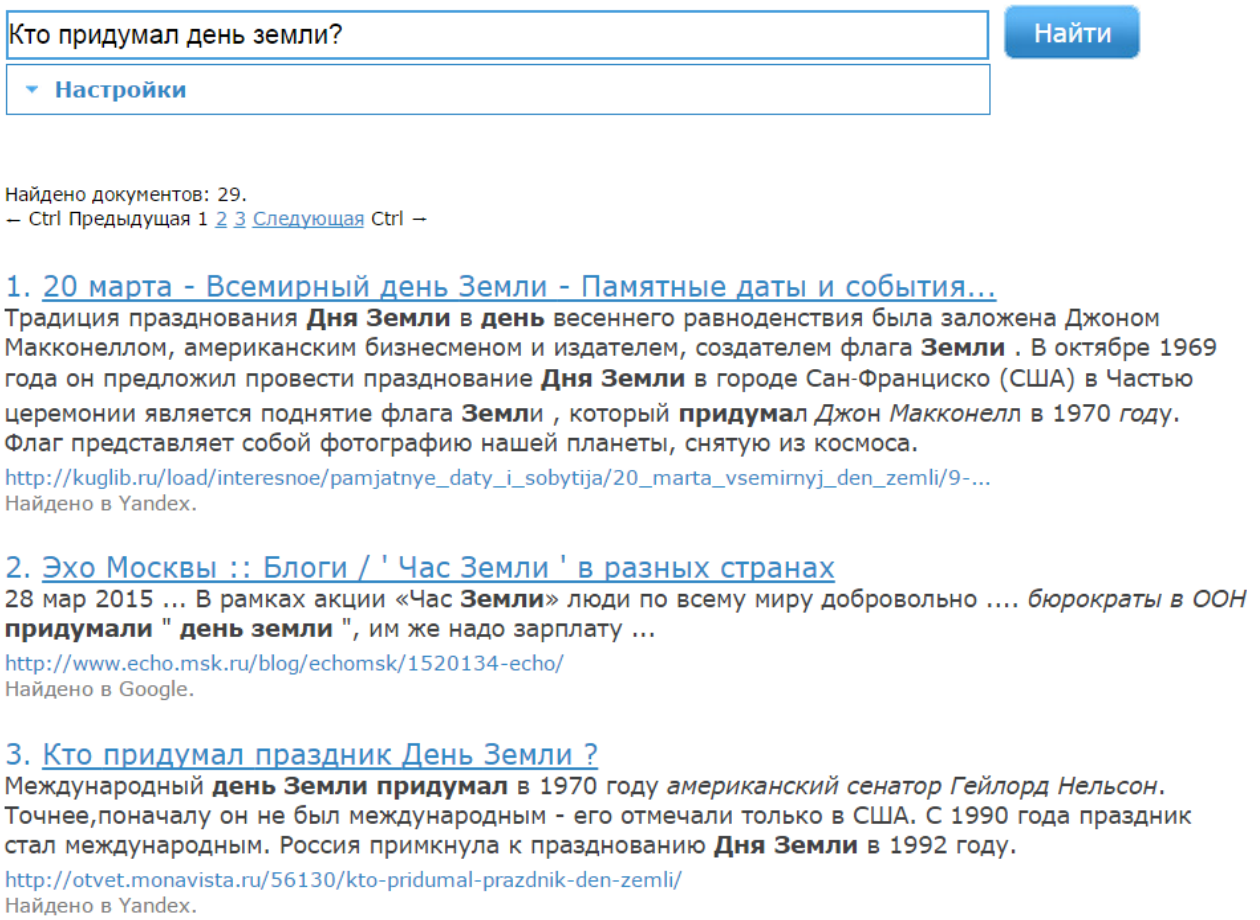


Рисунок 36 – Результат работы модуля метапоиска системы Exactus.ru

3) Качество ранжирования при использовании системы семантико-синтаксического анализа существенно выше, чем при использовании системы, в которой синтаксический и семантический анализ

выполняются раздельно – точность выше на 4% при $d = 1$, существенно выше и значение MRR. Повышение качества построения синтаксических деревьев зависимостей и определения ролевых структур высказываний за счет использования метода семантико-синтаксического анализа позволяет более точно проводить анализ как вопросительных запросов, так и поисковых сниппетов. На рисунке 37 представлен пример, в котором система семантико-синтаксического анализа исправила синтаксическое дерево в сниппете ответа на вопрос «*откуда в Россию привезли картофель?*». За счет этого была установлена семантическая роль «директив» и семантическое отношение «TRA», которые помогли поднять ответ на вопрос на первое место в выдаче.

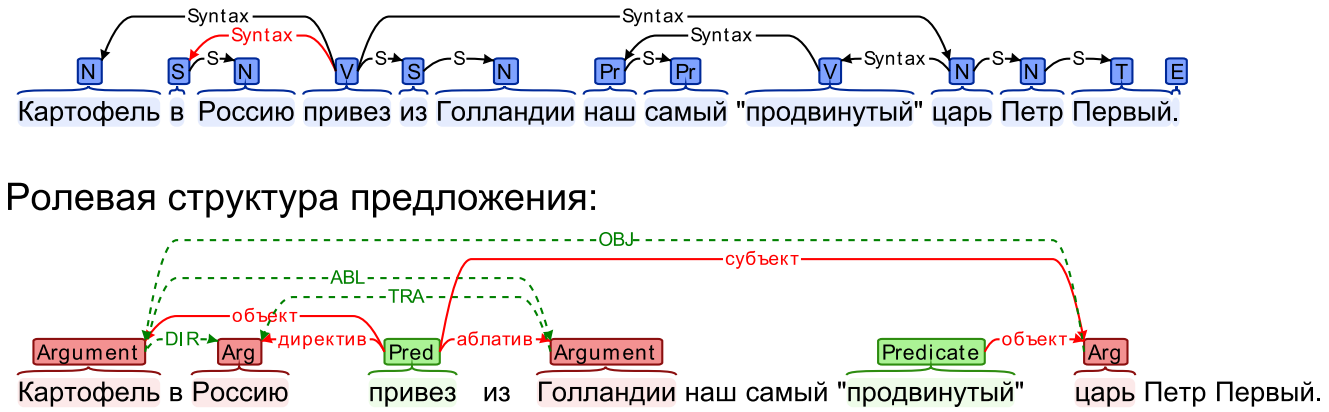


Рисунок 37 – Пример предложения в сниппете ответа на вопрос «откуда в Россию привезли картофель?», в котором система семантико-синтаксического анализа исправила синтаксическое дерево

4.3. Применение семантического и семантико-синтаксического анализа для решения задачи автоматического извлечения определений и авторских терминов из текстов научных публикаций

В научных публикациях авторы часто вводят новые термины или для удобства восприятия статьи дают определения (дефиниции) известным терминам. В этом разделе рассматривается задача извлечения из текстов научных публикаций определений и определяемых терминов, такие термины в дальнейшем будем называть авторскими.

В прикладных задачах информационного поиска определения и авторские термины, можно использовать для создания терминологической базы предметной области, пополнения тезаурусов, для формирования списка ключевых слов документа, его аннотации или, например, особым образом учитывать их в формуле расчета релевантности при поиске информации. Полнотекстовой поиск по определениям и авторским терминам может помочь пользователю понять смысл неизвестных терминов при эксплоративном поиске в коллекциях научных публикаций [172].

В этом разделе предлагается метод автоматического извлечения определений и авторских терминов из текстов научных публикаций, использующий морфологию, синтаксис и семантику, и описываются его экспериментальные исследования на вручную размеченном корпусе научных публикаций.

4.3.1. Метод извлечения определений и авторских терминов из текстов научных публикаций

При извлечении авторских терминов из научных публикаций надо учитывать, что в их текстах могут содержаться определения не только уже

устоявшихся, но и новых, ранее неизвестных терминов, сами термины могут содержать иностранные слова и словосочетания, а публикации могут быть из разных научных областей. Все это требует от метода извлечения определений и авторских терминов гибкости, например, подход, основанный на тезаурусах, в данном случае был бы малоэффективным.

Для решения поставленной задачи был предложен метод, основанный на сравнении лексико-синтаксической и семантической структуры предложения, со списком фреймов (шаблонов). Схожие подходы предлагались и в других работах. Например, в [173 – 176] описывается система, применяющая лексико-синтаксические шаблоны для извлечения авторских терминов различного типа из текстов научных публикаций. В настоящей работе было предложено помимо лексической, морфологической и синтаксической информации при построении фреймов использовать также семантические роли. Использование информации о семантических ролях позволяет, с одной стороны, упростить создание фреймов, а, с другой стороны, одним фреймом покрыть больше случаев определения терминов.

Экспертами был проанализирован корпус научных публикаций, ими было выделено более 60 устойчивых контекстов, наличие которых указывает на присутствие в предложении определения термина. На основе экспериментальных данных были уточнены контексты и сформирован набор фреймов (всего около 15), которые позволяют найти предложение, в котором присутствует определение термина и выделить подстроку, соответствующую авторскому термину. В таблице 10 приведено несколько примеров частей этих фреймов, позволяющие отыскать предложение, в котором присутствует определение термина. Во фреймах учитывается разная информация о тексте: лексика, морфология, синтаксис и семантика.

Таблица 10 – Примеры фреймов для нахождения определений и извлечения авторских терминов из текстов научных публикаций

Фрейм	**Примеры**
ЧР(Сущ.) && Падеж(Им.) + * + Л(«–») + Л(«этот») + * + ЧР(сущ.) && Падеж(Им.)	Дисперсия случайной величины – мера разброса данной случайной величины, то есть её отклонения от математического ожидания.
ЧР(Сущ.) && Сем.роль(эстиматив) + Л(«называться»)	Перигелием называется точка орбиты небесного тела, где оно максимально сближается с Солнцем.
Сем.роль.(делибератив) + Л(«определяться») + Л(«как»)	Философия традиционно определяется как исследование первопричин и начал всего сущего — универсальных принципов, в рамках которых существует и изменяется как бытие, так и мышление, как постигаемый Космос, так и постигающий его дух.
НФ(«под») + Сем.роль.(эстиматив) + НФ(«быть»)&&Время(буд.) + (Л(«понимать») || Л(«пониматься») || Л(«подразумевать») || Л(«подразумеваться»))	Под алгеброй мы будем понимать линейное пространство A (например, над полем действительных чисел R), снабженное ассоциативной операцией умножения и единицей,
Л(«представлять») + Л(«себя»)	Цепная ядерная реакция представляет собой деление ядра на две части, называемые осколками деления, с одновременным выделением нескольких (2—3) нейтронов, которые, в свою очередь, могут вызвать деление следующих ядер.
ЧР(Сущ.)&&Падеж(Им.) + Л(«–») + * + ЧР(Сущ.)&&Падеж(Им.)	Агенс – производитель действия (закон подписан президентом)
Обозначения: «Л» – лемма; «ЧР» – часть речи; «Сем.роль» – семантическая роль; && – логическое и; || – логическое или.	

На вход анализатора поступает структурированное представление текста, в котором выделены слова и предложения, указаны морфологические характеристики слов, включая леммы, в предложениях выделены клаузы, построены синтаксические деревья зависимостей и ролевые структуры предложений. Каждое предложение текста сопоставляется с фреймами из заданного списка, каждый из которых независимо от других извлекает один или несколько терминов. После этого извлеченные термины фильтруются набором эвристик. В результате, на выходе формируется список терминов, которым дано определение, и предложения, в которых встречаются эти термины. Предложения, в которых встречаются извлеченные термины помечаются как определения.

Процедура сопоставления фрейма с предложением текста состоит из двух этапов. На первом этапе в предложении ищутся «ключевые» слова фреймов. В каждом фрейме существует проверка наличия в предложении определенной леммы. Слово, имеющее эту лемму, будем называть «ключевым» для фрейма. Перед тем как проверять другие условия фреймов, осуществляется проход по предложению текста, в результате которого отыскиваются расположения всех ключевых слов и формируется список фреймов, для которых совпали ключевые слова. Такой подход удобен для последующего анализа, поскольку позволяет быстро отсеять предложения, не подпадающие ни под один фрейм, и сформировать сокращённый список фреймов, которые будут сопоставляться с заданным предложением. На следующем этапе для предложения проверяются другие условия фреймов, соответствующих найденным в нем ключевым словам. Предложения в научных текстах, как правило, имеют довольно сложную структуру, поэтому проверку удобно проводить не на всем предложении, а в рамках одной клаузы. Например, в предложении «*перигелием называется точка орбиты небесного тела, где оно максимально сближается с Солнцем*» будет сначала отдельно проанализирована часть «*перигелием называется точка орбиты небесного тела*», а затем часть «*где оно максимально сближается с*

Солнцем», и только первая часть будет помечена, как соответствующая фрейму 2.

После нахождения клаузы, удовлетворяющей некоторому фрейму, выделяется непосредственно авторский термин. Для его извлечения с каждым фреймом связано правило, которое определяет, какую часть подвыражения необходимо выделить в качестве термина. Это правило учитывает, что термин может состоять из нескольких слов. Например, в пятом фрейме из таблицы 10 термином будут считаться все слова, находящиеся в соответствующей клаузе и стоящие левее слова «представлять», а во втором фрейме синтаксическая группа, вершиной которой является слово, соответствующее аргументу с ролью «эстиматив».

Большинство терминов находится с помощью описанного способа поиска фреймов, однако, существует ряд распространенных случаев, которые вызывают ошибки при извлечении авторского термина:

- Выделенное выражение содержит слова, которые не являются терминами. В примере к третьему фрейму из таблицы 10: «Философия традиционно определяется как исследование первопричин и начал всего сущего…» таким словом будет «традиционно».
- Выражения с отрицанием, содержащие частицу «не». Пример: «но процесс загрузки не представляет собой навал, в реактор помещают тепловыделяющие элементы, уже хорошо нам известные ТВЭЛ».
- Выражения, которые задают отношения синонимии, часть-целое или некоторые случаи отношения «что является чем». Пример: «самым простым является топический параметр».

Чтобы учесть подобные случаи, после нахождения фрейма и выделения термина дополнительно проводится процедура фильтрации с помощью набора эвристик.

Для первого случая предусмотрены правила, которые фильтруют список слов, входящих в выделенный термин. Например, отфильтровываются некоторые частицы и союзы («пусть», «если», «также» и др.), наречия (традиционно, обыкновенно и др.) краткие прилагательные, стоящие непосредственно в конце и в начале многословного термина.

Чтобы отфильтровать выражения, подпадающие под второй случай, в соответствующей клаузе проверяется наличие у ключевого слова (если это глагол) синтаксической связи с частицей «не». Частично удается решить проблему из третьего случая путем фильтрации терминов по ключевым словам и формам, в которых эти слова могут находиться.

4.3.2. Экспериментальное исследование метода извлечения определений и авторских терминов из текстов научных публикаций

Для экспериментальной проверки разработанного метода был размечен корпус научных публикаций, содержащий статьи журналов из перечня ВАК и доклады на российских научных конференциях. Размер размеченного корпуса составляет более 72 000 токенов, в корпусе выделено более 300 определений. Рассчитывались нестрогие метрики: точность p, полнота r и F_1 мера. В нестрогих метриках пересечение отрезка текста в проверочном корпусе с отрезком, полученным автоматически от анализатора, даже без точного совпадения краев отрезков, считается правильным ответом. Сравнивались результаты трех анализаторов:

- Анализатор, в котором реализованы фреймы, использующие семантические роли, полученные от системы, в которой синтаксический и семантический анализ выполняются раздельно. Обозначим этот анализатор как «Сем».

- Анализатор, в котором реализованы фреймы, использующие семантические роли, полученные от системы семантико-синтаксического анализа. Обозначим этот анализатор как «Сем.-син.».
- Анализатор, в котором отсутствуют фреймы, использующие семантические роли. Обозначим этот анализатор как «Без сем.».

Результаты экспериментов представлены в таблице 11.

Таблица 11 – Оценки качества метода извлечения авторских терминов из текстов научных публикаций

Анализатор	***p*,%**	***r*,%**	***F_1*,%**
Сем	82,6	68,3	74,8
Сем.-син.	**82,7**	**68,7**	**75,0**
Без сем.	81,6	58,3	68,0

Полученные результаты показывают, что фреймы, использующие семантические роли обрабатывают существенное количество случаев определения терминов в научных публикациях (более 10% по полноте), тем самым вносят значительный вклад в решение поставленной задачи. Использование системы семантико-синтаксического анализа не дало значимого прироста качества по сравнению с использованием системы, в которой синтаксический и семантический анализ выполняются раздельно. Причина этого заключается в том, что наибольшую роль в решении этой задачи играют лексико-морфологические правила, не учитывающие семантику и синтаксис предложений, следовательно, прирост полноты определения ролевых структур не сильно отражается на конечном результате.

Анализ ошибок выявил, что значительное число терминов, которые не были извлечены, требуют для их нахождения разрешение кореферентных связей как на уровне одного предложения, так и на уровне всего текста. Другая значительная часть ненайденных терминов, размеченных в проверочном корпусе,

соответствуют отдельным редким паттернам. На точность негативно влияют ошибки фрейма, где определяемый термин и его определение разделены тире или дефисом. Фрейм, который бы соответствовал лишь определениям в таких случаях, выделить довольно трудно из-за большого количества различных лингвистических конструкций, использующих тире или дефис, а также ошибок их употребления в ЕЯ текстах.

В целом полученные результаты иллюстрируют применимость разработанного анализатора, в котором реализованы правила, использующие семантические роли, полученные в результате семантико-синтаксического анализа, для решения задачи извлечения определений и авторских терминов в прикладных приложениях.

Модуль извлечения определений и авторских терминов используется в функции полнотекстового поиска по определениям в информационно-аналитической системе Exactus Expert[7]. Извлеченные определения и термины помечаются в поисковом индексе отдельными тегами. Поисковый механизм, позволяет задавать ограничение зоны поиска по таким тегам. Благодаря этому пользователи могут совершать поиск не по полным текстам статей, а по терминам, для которых есть определение. Это позволяет пользователю быстро отыскать непосредственно определение неизвестного термина, что является весьма полезным при эксплоративном поиске. На рисунке 38 представлен пример работы функции полнотекстового поиска по определениям в системе Exactus Expert.

[7] http://expert.exactus.ru/

1. О полных односторонних функциях

...Мы не будем давать отдельных определений, а будем считать, что недетерминированная **машина Тьюринга** – это обычная детерминированная машина Тьюринга с двумя лентами, на которых записаны две части входа: собственно вход и “подсказка”; недетерминированная машина Тьюринга M вычисляет функцию f : Σ ∗ → Σ ∗ , если для каждого входа x существует такая “подсказка” y ∈ Σ ∗ , что M (x, y) = f (x), и ни для какой подсказки машина не выдает неверный ответ (разрешается, однако, завершать вычисление, не выдавая никакого ответа). <...>...

Авторы: А. А. Кожевников , С. И. Николенко.

Год публикации: 2009.

http://www.mathnet.ru/links/583ba2668dc0f297505f736a18ed192c/ppi1982.pdf

Похожие | Резюме | Ключевые слова | Оценка качества

2. СЛОЖНОСТЬ ВЫЧИСЛЕНИЙ НА АБСТРАКТНЫХ МАШИНАХ

...Тьюринг избрали форму такого описания в виде абстрактных машин; они теперь называются **машинами Тьюринга.** <...> Однако в настоящее время подобные вычислительные средства называются **машинами Тьюринга.** <...>...

Авторы: ЛАВРОВ И.А..

Год публикации: 2007.

http://cyberleninka.ru/article/n/slozhnost_vychisleniy_na_abstraktnyh_mashinah

Похожие | Резюме | Ключевые слова | Оценка качества

3. Моделирование схем из функциональных элементов машинами Тьюринга

...Под **машиной Тьюринга** M будем понимать конечный автомат A с к односторонними лентами A b . . . , A j b , состо ящими из ячеек, пронумерованных слева направо натуральными чис лами. <...>...

Авторы: А. В. Чашкин.

Год публикации: 1999.

http://www.mathnet.ru/links/b374ffc39445f34df224ff74583e4a80/da321.pdf

Похожие | Резюме | Ключевые слова | Оценка качества

Рисунок 38 – Поисковая выдача, полученная в результате поиска по определениям по запросу «{машина тьюринга}». В тексте поисковых сниппетов подсвечены ключевые слова из запроса

4.4. Выводы

Продемонстрирована применимость разработанных методов семантического и семантико-синтаксического анализа для решения прикладных задач обработки текстов на естественном языке.

Показано, как информация о семантических ролях, полученная от семантического и семантико-семантического анализаторов, применяется в

алгоритме построения семантической сети для реляционно-ситуационной модели текста.

Разработан метод ранжирования сниппетов для вопросно-ответного поиска в метапоисковой системе, в котором наряду с лексикой, учитываются семантические роли и отношения. Экспериментально показано, что семантическая информация вносит большой вклад в точность работы алгоритма ранжирования и позволяет извлекать из сниппетов непосредственно ответы на вопросы. Кроме этого, результаты экспериментальных исследований свидетельствуют о значительном преимуществе использования системы семантико-синтаксического анализа при решении задачи вопросно-ответного поиска по сравнению с системой, в которой синтаксический и семантический анализ выполняются раздельно.

Разработан метод извлечения определений и авторских терминов из текстов научных публикаций, в котором реализованы правила учитывающие семантические роли. Экспериментально показана эффективность разработанного метода и значимость вклада семантических правил в решение этой задачи. Использование ролевой структуры предложения упрощает построение правил для извлечения определений и авторских терминов.

Методы проведения семантического и семантико-синтаксического анализа, метод построения семантической сети предложения, метод ранжирования сниппетов для вопросно-ответного поиска, а также метод извлечения определений и определяемых терминов реализованы и внедрены в поисковые и поисково-аналитические системы Exactus, Exactus Expert, а также в коммерческие электронные библиотечные системы РУКОНТ[8] и Znanium.com[9].

[8] http://rucont.ru/

[9] http://search.znanium.com/

ЗАКЛЮЧЕНИЕ

Перечислим основные результаты, полученные в диссертационном исследовании:

1. Разработан новый метод автоматического определения ролевых структур высказываний, основанный на коммуникативной грамматике русского языка.
2. Разработан новый метод компьютерного семантико-синтаксического анализа текстов, в котором интегрированы методы построения синтаксических деревьев зависимостей и определения ролевых структур высказываний, позволяющий повысить точность и полноту синтаксического и семантического анализа по сравнению с реализацией, в которой эти виды анализа выполняются раздельно.
3. Разработана и реализована интегрированная система семантико-синтаксического анализа. Система применена для решения задач вопросно-ответного поиска, извлечения определений и авторских терминов из текстов научных публикаций.
4. Экспериментально показано, что при использовании интегрированной системы семантико-синтаксического анализа существенно повышается точность вопросно-ответного поиска по сравнению с раздельным применением методов синтаксического и семантического анализа.
5. Экспериментально показано, что использование ролевой структуры предложения повышает эффективность построения правил для извлечения определений и авторских терминов из текстов научных публикаций.

СПИСОК СОКРАЩЕНИЙ И УСЛОВНЫХ ОБОЗНАЧЕНИЙ

ЕЯ	Естественный язык
КС-грамматика	Контекстно-свободная грамматика
КС-язык	Контекстно-свободный язык
Алгоритм CYK	Алгоритм Cocke – Younger – Kasami
GLR-анализатор	Generalized Left-to-right Rightmost анализатор
СКС-грамматика	Стохастическая контекстно-свободная грамматика
ПС	Предикатное слово
ПА-структура	Предикатно-аргументная структура
КСК	Категориально-семантический класс
ЛЦП	Линейное целочисленное программирование
UAS	Unlabeled attachment score
СР	Семантическая роль
СО	Семантическое отношение
MRR	Mean reciprocal rank

СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ

1 Смирнов И. В., Шелманов А. О. Семантико-синтаксический анализ естественных языков. Часть I. Обзор методов синтаксического и семантического анализа текстов // Искусственный интеллект и принятие решений. — 2013. — № 1. — С. 41–54.

2 Relational–situational method for intelligent search and analysis of scientific publications / Gennady Osipov, Ivan Smirnov, Ilya Tikhomirov, Artem Shelmanov // Proceedings of the Workshop on Integrating IR technologies for Professional Search, in conjunction with the 35th European Conference on Information Retrieval (ECIR'13). — Vol. 968. — CEUR Workshop Proceedings, 2013.

3 Семантико-синтаксический анализ естественных языков Часть II. Метод семантико-синтаксического анализа текстов / И. В. Смирнов, А. О. Шелманов, Е. С. Кузнецова, И. В. Храмоин // Искусственный интеллект и принятие решений. — № 1. — С. 11–24.

4 Shelmanov A. O., Smirnov I. V. Methods for semantic role labeling of Russian texts // Computational Linguistics and Intellectual Technologies. Papers from the Annual International Conference "Dialogue" (2014). — No. 13. — 2014. — P. 607–620.

5 Осипов Г. С., Шелманов А. О. Метод повышения качества синтаксического анализа на основе взаимодействия синтаксических и семантических правил // Труды шестой международной конференции "Системный анализ и информационные технологии" (САИТ). — Т. 1. — 2015. — С. 229–240.

6 Шелманов А. О. Метод автоматического выделения многословных терминов из текстов научных публикаций // Труды тринадцатой национальной конференции по искусственному интеллекту с международным участием КИИ-2012. — Т. 1. — Белгород, 2012. — С. 268–274.

7 Шелманов А. О., Смирнов И. В. «Программа лингвистического анализа неструктурированной текстовой информации на русском и английском языках» // Свидетельство о государственной регистрации программ для ЭВМ. № 2013613430. — 2013.

8 ETAP parser: state of the art / L. Iomdin, V. Petrochenkov, V. Sizov, L. Tsinman // Papers from the Annual International Conference "Dialogue" (2012). — 2012. — P. 830–853.

9 Syntactic and semantic parser based on ABBYY Compreno linguistic technologies / K. V. Anisimovich, K. Ju. Druzhkin, F. R. Minlos et al. // Papers from the Annual International Conference "Dialogue" (2012). — Vol. 2. — 2012. — P. 91–103.

10 Chomsky N. Three models for the description of language // IRE Transactions on Information Theory. — 1956. — Vol. 2, no. 3. — P. 113–124.

11 Tesnière L. Elements de syntaxe structurale. — Editions Klincksieck, 1959.

12 Mel'cuk I. A. Dependency syntax: theory and practice. — ŠUNY Press, 1988. — P. 428.

13 Chomsky N. Syntactic structures. — The Hague : Mouton, 1957. — P. 117.

14 Partee B., Ter Meulen A., Wall R. Mathematical methods in linguistics. — Springer, 1990. — Vol. 30 of Studies in Linguistics and Philosophy.

15 Huybregts R. The weak inadequacy of context-free phrase structure grammars // Van Periferie naar Kern. — 1984. — P. 81–99.

16 Shieber S. M. Evidence against the context-freeness of natural language // The Formal Complexity of Natural Language. — 1987. — Vol. 33. — P. 320–334.

17 Hudson R. Word grammar. — Blackwell Oxford, 1984.

18 Karlsson F. Constraint grammar as a framework for parsing running text // Proceedings of the 13th conference on Computational linguistics. — Vol. 3. — Association for Computational Linguistics, 1990. — P. 168–173.

19 Universal dependency annotation for multilingual parsing / Ryan Mcdonald, Joakim Nivre, Yvonne Quirmbach-brundage et al. // Proceedings of the 51st Annual

Meeting of the Association for Computational Linguistics. — Vol. 2. — 2013. — P. 92–97.

20 Nivre J. Dependency grammar and dependency parsing // MSI report. — 2005. — Vol. 5133, no. 1959. — P. 1–32.

21 Non-projective dependency parsing using spanning tree algorithms / Ryan McDonald, Fernando Pereira, Kiril Ribarov, Jan Hajic // Proceedings of the conference on Human Language Technology and Empirical Methods in Natural Language Processing. — Association for Computational Linguistics, 2005. — P. 523–530.

22 Тестелец Я. Г. Введение в общий синтаксис. — М.: Издательство РГГУ, 2001.

23 Jurafsky D., Martin J. H. Speech & language processing. — Pearson Education, 2000.

24 Сулейманова Е. А. О комплексном подходе к разрешению реляционно-аппозитивных неоднозначностей // Программные системы: теория и приложения. — 2014. — Т. 5, № 4. — С. 41–66.

25 Kasami T. An efficient recognition and syntax analysis algorithm for context-free languages. — 1965. — Technical report.

26 Younger D. H. Recognition and parsing of context-free languages in time n^3 // Information and control. — 1967. — Vol. 10, no. 2. — P. 189–208.

27 Earley J. An efficient context-free parsing algorithm // Communications of the ACM. — 1970. — Vol. 13, no. 2. — P. 94–102.

28 Kay M. Algorithm schemata and data structures in syntactic processing // Technical Report CSL80-12. — 1980.

29 Tomita M. LR parsers for natural languages // Proceedings of the 10th international conference on Computational linguistics. — Association for Computational Linguistics, 1984. — P. 354–357.

30 Collins M. Head-driven statistical models for natural language parsing // Computational linguistics. — 2003. — Vol. 29, no. 4. — P. 589–637.

31 Marcus M. P., Marcinkiewicz M. A., Santorini B. Building a large annotated corpus of English the Penn treebank // Computational Linguistics. — 1993. — Vol. 19, no. 2. — P. 313–330.

32 Collins M. Three generative, lexicalised models for statistical parsing // Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics and Eighth Conference of the European Chapter of the Association for Computational Linguistics. — Association for Computational Linguistics, 1997. — P. 16–23.

33 Hays D. G. Dependency theory: A formalism and some observations // Language. — 1964. — Vol. 40. — P. 511–525.

34 Gaifman H. Dependency systems and phrase-structure systems // Information and control. — 1965. — Vol. 8, no. 3. — P. 304–337.

35 Holan T., Kubon V., Plátek M. A prototype of a grammar checker for Czech // Proceedings of the fifth conference on Applied natural language processing. — Association for Computational Linguistics, 1997. — P. 147–154.

36 Lombardo V., Lesmo L. An Earley-type recognizer for dependency grammar // Proceedings of the 16th conference on Computational linguistics. — Association for Computational Linguistics, 1996. — P. 723–728.

37 Maruyama H. Structural disambiguation with constraint propagation // Proceedings of the 28th annual meeting on Association for Computational Linguistics. — Association for Computational Linguistics, 1990. — P. 31–38.

38 Covington M. A. A fundamental algorithm for dependency parsing // Proceedings of the 39th annual ACM Southeast Conference. — 2001. — P. 95–102.

39 Eisner J. M. Three new probabilistic models for dependency parsing: An exploration // Proceedings of the 16th conference on Computational linguistics. — Vol. 1. — Association for Computational Linguistics, 1996. — P. 340–345.

40 McDonald R., Crammer K., Pereira F. Online large-margin training of dependency parsers // Proceedings of the 43rd annual meeting on Association for Computational Linguistics. — Association for Computational Linguistics, 2005. — P. 91–98.

41 Chu Y.-J., Liu T.-H. On shortest arborescence of a directed graph // Scientia Sinica. — 1965. — Vol. 14, no. 10. — P. 1396.

42 Edmonds J. Optimum branchings // Journal of Research of the National Bureau of Standards, Section B: Mathematics and Mathematical Physics. — 1967. — Vol. 71B, no. 4. — P. 233–240.

43 Mcdonald R., Pereira F. Online learning of approximate dependency parsing algorithms // Proceedings of European Chapter of the Association for Computational Linguistics. — Association for Computational Linguistics, 2006.

44 Nakagawa T. Multilingual dependency parsing using global features // Proceedings of the CoNLL Shared Task Session of EMNLP-CoNLL 2007. — Association for Computational Linguistics, 2007. — P. 952–956.

45 Carreras X. Experiments with a higher-order projective dependency parser // Proceedings of the CoNLL Shared Task Session of EMNLP-CoNLL 2007. — Association for Computational Linguistics, 2007. — P. 957–961.

46 Zhang H., McDonald R. Generalized higher-order dependency parsing with cube pruning // Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning. — Association for Computational Linguistics, 2012. — P. 320–331.

47 McDonald R., Satta G. On the complexity of non-projective data-driven dependency parsing // Proceedings of the 10th International Conference on Parsing Technologies. — Association for Computational Linguistics, 2007. — P. 121–132.

48 Kudo T., Matsumoto Y. Japanese dependency structure analysis based on support vector machines // Proceedings of the 2000 Joint SIGDAT conference on Empirical methods in natural language processing and very large corpora: held in conjunction

with the 38th Annual Meeting of the Association for Computational Linguistics. — Vol. 13. — Association for Computational Linguistics, 2000. — P. 18–25.

49 Yamada H., Matsumoto Y. Statistical dependency analysis with support vector machines // Proceedings of 8th International Workshop on Parsing Technologies. — 2003. — P. 195–206.

50 Nivre J., Hall J., Nilsson J. Memory-based dependency parsing // Proceedings of CoNLL. — 2004. — P. 49–56.

51 Nivre J. Inductive Dependency Parsing of Natural Language Text : Ph.D. thesis / Joakim Nivre ; School of Mathematics and Systems Engineering, Växjö University. — 2005.

52 Nivre J., Nilsson J. Pseudo-projective dependency parsing // Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics. — Association for Computational Linguistics, 2005. — P. 99–106.

53 MaltParser: A language-independent system for data-driven dependency parsing / Joakim Nivre, Johan Hall, Jens Nilsson et al. // Natural Language Engineering. — 2007. — Vol. 13, no. 2. — P. 95–135.

54 Nivre J., Hall J., Nilsson J. MaltParser: A data-driven parser-generator for dependency parsing // Proceedings of the International Conference on Language Resources and Evaluation (LREC). — Vol. 6. — 2006. — P. 2216–2219.

55 Nivre J., Boguslavsky I. M., Iomdin L. L. Parsing the SynTagRus treebank of Russian // Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008). — Manchester, UK, 2008. — August. — P. 641–648.

56 Sharoff S., Nivre J. The proper place of men and machines in language technology: Processing Russian without any linguistic knowledge // Papers from the Annual International Conference "Dialogue" (2011). — No. 10. — 2011. — P. 17.

57 Апресян Ю. Д. Лингвистическое обеспечение системы ЭТАП-2. — Наука, 1989.

58 Казенников А. О. Сравнительный анализ статистических алгоритмов синтаксического анализа на основе деревьев зависимостей // Труды международной конференции "Диалог 2010". — 2010.

59 Белоногов Г. Г., Калинин Ю. П., Хорошилов А. А. Компьютерная лингвистика и перспективные информационные технологии. — М: Русский мир, 2004.

60 Белоногов Г. Г. Теоретические проблемы информатики, том 2. семантические проблемы информатики // М.: РЭА им. Г.В. Плеханова. — 2008.

61 Белоногов Г. Г., Гиляровский Р. С., др. Проблемы автоматической смысловой обработки текстовой информации // Научно-техническая информация. Сер. 2. Информационные процессы и системы. — 2012. — № 11. — С. 24–28.

62 Ножов И. М. Морфологическая и синтаксическая обработка текста (модели и программы) : Дисс кандидата наук / И. М. Ножов. — 2003.

63 Автоматизация отладки алгоритмов поверхностно-синтаксического анализа / А. М. Баталина, Г. Ю. Айриян, М. Е. Епифанов и др. // Труды международной конференции "Диалог 2005". — 2005. — С. 45–50.

64 Кобзарева Т. Ю., Епифанов М. Е., Лахути Д. Г. Восстановление грамматических эллипсисов при синтаксическом анализе // Труды международной конференции КИИ-2014. — 2014.

65 Каневский Е. А., Боярский К. К. Семантико-синтаксический анализатор SemSin // Труды международной конференции "Диалог 2012". — 2012.

66 Antonova A. A., Misyurev A. V. Russian dependency parser SyntAutom at the DIALOGUE-2012 parser evaluation task // Papers from the Annual International Conference "Dialogue" (2012). — 2012.

67 Синтаксически и семантически аннотированный корпус русского языка: современное состояние и перспективы / Ю. Д. Апресян, И. М. Богуславский, Б. Л. Иомдин и др. // Национальный корпус русского языка: 2003–2005. — 2005. — С. 193–214.

68 Поспелов Д. А. Десять горячих точек в исследованиях по искусственному интеллекту // Интеллектуальные системы (МГУ). — 1996. — Т. 1, № 1-4. — С. 47–56.

69 Jurafsky D., Martin J. Speech And Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition. Prentice Hall Series in Artificial Intelligence. — Pearson Prentice Hall, 2009.

70 Montague R. The proper treatment of quantification in ordinary English // Approaches to Natural Language / Ed. by K. J. J. Hintikka, J. Moravcsic, P. Suppes. — Springer, 1973. — P. 221–242.

71 Bach E. An extension of classical transformational grammar // Problems of Linguistic Metatheory (Proceedings of the 1976 Conference). — 1976.

72 Nishida T.-a., Doshita S. An English-Japanese machine translation system based on formal semantics of natural language // Proceedings of the 9th conference on Computational linguistics. — Vol. 1. — 1982. — P. 277–282.

73 Fillmore C. J. The case for case // Universals in Linguistic Theory / Ed. by Emmon Bach, Robert T. Harms. — New York, 1968. — P. 1–88.

74 Осипов Г. С. Методы искусственного интеллекта. — ФИЗМАТЛИТ, 2011.

75 Jackendoff R. Semantic interpretation in generative grammar. — MIT press Cambridge, MA, 1972.

76 Богданов В. В. Семантико-синтаксическая организация предложения. — Изд-во Ленинградского университета, 1977.

77 Кибрик А. Е. Очерки по общим и прикладным вопросам языкознания: (универсальное, типовое и специфическое в языке). — М.: Изд-во МГУ, 1992.

78 Кашкин Е. В., Ляшевская О. Н. Семантические роли и сеть конструкций в системе FrameBank // Труды международной конференции "Диалог 2013". — 2013. — С. 325–343.

79 Плунгян В. А. Введение в грамматическую семантику: грамматические значения и грамматические системы языков мира: учебное пособие. — М.: Издательство РГГУ, 2011.

80 Dowty D. Thematic proto-roles and argument selection // Language. — 1991. — P. 547–619.

81 Palmer M., Gildea D., Kingsbury P. The proposition bank: An annotated corpus of semantic roles // Computational linguistics. — 2005. — Vol. 31, no. 1. — P. 71–106.

82 Fillmore C. J., Johnson C. R., Petruck M. R. Background to FrameNet // International journal of lexicography. — 2003. — Vol. 16, no. 3. — P. 235–250.

83 The NomBank project: An interim report / A. Meyers, R. Reeves, C. Macleod et al. // HLT-NAACL 2004 Workshop: Frontiers in Corpus Annotation. — Association for Computational Linguistics, 2004. — P. 24–31.

84 Xue N., Palmer M. Adding semantic roles to the Chinese treebank // Natural Language Engineering. — 2009. — Vol. 15, no. 1. — P. 143–172.

85 The revised Arabic PropBank / Wajdi Zaghouani, Mona Diab, Aous Mansouri et al. // Proceedings of the Fourth Linguistic Annotation Workshop. — Association for Computational Linguistics, 2010. — P. 222–226.

86 Using FrameNet for the semantic analysis of German: Annotation, representation, and automation / Aljoscha Burchardt, Katrin Erk, Anette Frank et al. // Multilingual FrameNets in Computational Lexicography / Ed. by Hans C. Boas. — Mouton, 2009. — P. 209–244.

87 Subirats C. Spanish FrameNet: A frame-semantic analysis of the Spanish lexicon // Multilingual FrameNets in Computational Lexicography: Methods and Applications / Ed. by Hans Boas. — Mouton de Gruyter, 2009. — P. 135–162.

88 Gildea D., Jurafsky D. Automatic labeling of semantic roles // Computational Linguistics. — 2002. — Vol. 28, no. 3. — P. 245–288.

89 Carreras X., Marquez L. Introduction to the CoNLL-2004 shared task: Semantic role labeling // Proceedings of CoNLL'2004 Shared Task. — 2004.

90 Carreras X., Màrquez L. Introduction to the CoNLL-2005 shared task: Semantic role labeling // Proceedings of the Ninth Conference on Computational Natural Language Learning (CoNLL-2005). — Association for Computational Linguistics, 2005. — P. 152–164.

91 The CoNLL-2008 shared task on joint parsing of syntactic and semantic dependencies / Mihai Surdeanu, Richard Johansson, Adam Meyers et al. // Proceedings of the Twelfth Conference on Computational Natural Language Learning. — Association for Computational Linguistics, 2008. — P. 159–177.

92 The CoNLL-2009 shared task: Syntactic and semantic dependencies in multiple languages / Jan Hajic, Massimiliano Ciaramita, Richard Johansson et al. // Proceedings of the Thirteenth Conference on Computational Natural Language Learning: Shared Task. — Association for Computational Linguistics, 2009. — P. 1–18.

93 Shen D., Lapata M. Using semantic roles to improve question answering // Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL). — Association for Computational Linguistics, 2007. — P. 12–21.

94 Using predicate-argument structures for information extraction / Mihai Surdeanu, Sanda Harabagiu, John Williams, Paul Aarseth // Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics. — Association for Computational Linguistics, 2003. — P. 8–15.

95 Khan A., Salim N., Kumar Y. J. A framework for multi-document abstractive summarization based on semantic role labelling // Applied Soft Computing. — 2015. — Vol. 30. — P. 737–747.

96 Description of SQUASH, the SFU question answering summary handler for the DUC-2005 summarization task / G. Melli, Y. Wang, Y. Liu et al. — 2005.

97 Wu D., Fung P. Semantic roles for SMT: a hybrid two-pass model // Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics, Companion Volume: Short Papers. — Association for Computational Linguistics, 2009. — P. 13–16.

98 Liu D., Gildea D. Semantic role features for machine translation // Proceedings of the 23rd International Conference on Computational Linguistics. — Association for Computational Linguistics, 2010. — P. 716–724.

99 Osipov G. Methods for extracting semantic types of natural language statements from texts // 10th IEEE International Symposium on Intelligent Control. — Monterey, California, USA, 1995.

100 Золотова Г. А., Онипенко Н. К., Сидорова М. Ю. Коммуникативная грамматика русского языка // Институт русского языка РАН им. В. В. Виноградова. — 2004.

101 Осипов Г. С., Смирнов И. В., Тихомиров И. А. Реляционно-ситуационный метод поиска и анализа текстов и его приложения // Искусственный интеллект и принятие решений. — 2008. — № 2. — С. 3–10.

102 Osipov G., Smirnov I., Tikhomirov I. Relational-situational method for text search and analysis and its applications // Scientific and Technical Information Processing. — 2010. — Vol. 37, no. 6. — P. 432–437.

103 Application of linguistic knowledge to search precision improvement / Gennady Osipov, Ivan Smirnov, Ilya Tikhomirov, Olga Zavjalova // Intelligent Systems, 2008. IS'08. 4th International IEEE Conference. — Vol. 2. — 2008. — P. 17–2–17–5.

104 Shallow semantic parsing using support vector machines / Sameer S Pradhan, Wayne H Ward, Kadri Hacioglu et al. // HLT-NAACL 2004: Main Proceedings. — Association for Computational Linguistics, 2004. — P. 233–240.

105 Semantic role labeling using maximum entropy model / Joon-Ho Lim, Young-Sook Hwang, So-Young Park, Hae-Chang Rim // Proceedings of CoNLL. — 2004. — P. 122–125.

106 Xue N., Palmer M. Calibrating features for semantic role labeling // Proceedings of EMNLP 2004. — Association for Computational Linguistics, 2004. — P. 88–94.

107 Toutanova K., Haghighi A., Manning C. D. Joint learning improves semantic role labeling // Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics. — Association for Computational Linguistics, 2005. — P. 589–596.

108 Choi J. D., Palmer M. Transition-based semantic role labeling using predicate argument clustering // Proceedings of the ACL 2011 Workshop on Relational Models of Semantics. — Association for Computational Linguistics, 2011. — P. 37–45.

109 Johansson R., Nugues P. Dependency-based syntactic-semantic analysis with PropBank and NomBank // Proceedings of the Twelfth Conference on Computational Natural Language Learning. — Association for Computational Linguistics, 2008. — P. 183–187.

110 A high-performance syntactic and semantic dependency parser / Anders Björkelund, Bernd Bohnet, Love Hafdell, Pierre Nugues // Proceedings of the 23rd International Conference on Computational Linguistics: Demonstrations. — Association for Computational Linguistics, 2010. — P. 33–36.

111 Miller G. A. WordNet: A lexical database for English // Communications of the ACM. — 1995. — Vol. 38, no. 1. — P. 39–41.

112 Support vector learning for semantic argument classification / Sameer Pradhan, Kadri Hacioglu, Valerie Krugler et al. // Machine Learning. — 2005. — Vol. 60, no. 1-3. — P. 11–39.

113 Semantic role labeling: an introduction to the special issue / Lluis Marquez, Xavier Carreras, Kenneth Litkowski, Suzanne Stevenson // Computational linguistics. — 2008. — Vol. 34, no. 2. — P. 145–159.

114 Punyakanok V., Roth D., Yih W.-t. The importance of syntactic parsing and inference in semantic role labeling // Computational Linguistics. — 2008. — Vol. 34, no. 2. — P. 257–287.

115 Toutanova K., Haghighi A., Manning C. D. A global joint model for semantic role labeling // Computational Linguistics. — 2008. — Vol. 34, no. 2. — P. 161–191.

116 Thompson C. A., Levy R., Manning C. D. A generative model for semantic role labeling // Machine Learning: ECML 2003. — Springer, 2003. — P. 397–408.

117 Cohn T., Blunsom P. Semantic role labelling with tree conditional random fields // Proceedings of the Ninth Conference on Computational Natural Language Learning. — Association for Computational Linguistics, 2005. — P. 169–172.

118 Meza-Ruiz I., Riedel S. Jointly identifying predicates, arguments and senses using Markov logic // Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics. — Association for Computational Linguistics, 2009. — P. 155–163.

119 Combination strategies for semantic role labeling / Mihai Surdeanu, Lluís Màrquez, Xavier Carreras, Pere R Comas // Journal of Artificial Intelligence Research. — 2007. — P. 105–151.

120 Semantic role chunking combining complementary syntactic views / Sameer Pradhan, Kadri Hacioglu, Wayne Ward et al. // Proceedings of the Ninth Conference on Computational Natural Language Learning. — Association for Computational Linguistics, 2005. — P. 217–220.

121 Semantic role labeling as sequential tagging / Llus Màrquez, Pere Comas, Jesús Giménez, Neus Catala // Proceedings of the Ninth Conference on Computational Natural Language Learning. — Association for Computational Linguistics, 2005. — P. 193–196.

122 Lyashevskaya O. N., Kashkin E. V. Evaluation of frame-semantic role labeling in a case-marking language // Papers from the Annual International Conference "Dialogue" (2014). — 2014. — P. 350–365.

123 Ермаков А. Е., Плешко В. В. Семантическая интерпретация в системах компьютерного анализа текста // Информационные технологии. — 2009. — Т. 6. — С. 2–7.

124 Сокирко А. В. Семантические словари в автоматической обработке текста (по материалам системы ДИАЛИНГ) : Дисс кандидата наук / А. В. Сокирко. — 2001.

125 Смирнов И. В. Метод автоматического установления значений минимальных синтаксических единиц текста // Информационные технологии и вычислительные системы. — 2008. — № 3. — С. 30–45.

126 Padó S., Lapata M. Cross-lingual annotation projection for semantic roles // Journal of Artificial Intelligence Research. — 2009. — Vol. 36. — P. 307–340.

127 Johansson R., Nugues P. A FrameNet-based semantic role labeler for Swedish // Proceedings of the COLING/ACL. — Association for Computational Linguistics, 2006. — P. 436–443.

128 Fung P., Chen B. BiFrameNet: bilingual frame semantics resource construction by cross-lingual induction // Proceedings of the 20th international conference on Computational Linguistics. — Association for Computational Linguistics, 2004.

129 Kaljahi R., Foster J., Roturier J. Semantic role labelling with minimal resources: Experiments with French. — 2014. — P. 87–92.

130 Burchardt A., Erk K., Frank A. A WordNet Detour to FrameNet. — 2005.

131 Johansson R., Nugues P. Using WordNet to extend FrameNet coverage. — 2007.

132 Automatic induction of FrameNet lexical units / Marco Pennacchiotti, Diego De Cao, Roberto Basili et al. // Proceedings of the 2008 Conference on Empirical Methods in Natural Language Processing. — Association for Computational Linguistics, 2008.

133 Fürstenau H., Lapata M. Semi-supervised semantic role labeling via structural alignment // Computational Linguistics. — 2012. — Vol. 38, no. 1. — P. 135–171.

134 Lang J., Lapata M. Similarity-driven semantic role induction via graph partitioning // Computational linguistics. — 2014. — Vol. 40, no. 3. — P. 633–669.

135 Lang J., Lapata M. Unsupervised semantic role induction via split-merge clustering // Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies. — Vol. 1. — Association for Computational Linguistics, 2011. — P. 1117–1126.

136 Titov I., Klementiev A. Crosslingual induction of semantic roles // Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics. — Association for Computational Linguistics, 2012.

137 Baroni M., Zamparelli R. Nouns are vectors, adjectives are matrices: Representing adjective-noun constructions in semantic space // Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing. — Association for Computational Linguistics, 2010. — P. 1183–1193.

138 Ge R., Mooney R. J. A statistical semantic parser that integrates syntax and semantics // Proceedings of the Ninth Conference on Computational Natural Language Learning. — Association for Computational Linguistics, 2005. — P. 9–16.

139 Zettlemoyer L. S., Collins M. Learning to map sentences to logical form: Structured classification with probabilistic categorial grammars // In Proceedings of the 21st Conference on Uncertainty in AI. — 2005. — P. 658–666.

140 Tsuruoka Y., Miyao Y., Tsujii J. Towards efficient probabilistic HPSG parsing: integrating semantic and syntactic preference to guide the parsing // Proceedings of the IJCNLP-04 Workshop on Beyond Shallow Analyses. — 2004.

141 Levine R. D., Meurers W. D. Head-driven phrase structure grammar: Linguistic approach, formal foundations, and computational realization / Ed. by Keith Brown.

— Oxford : Elsevier, 2006. — Vol. 5. — P. 237–252. — URL: http://purl.org/dm/papers/ell2-hpsg.html.

142 Тузов В. А. Компьютерная семантика русского языка. — СПбГУ, 2003. — С. 400.

143 Lluìs X., Màrquez L. A joint model for parsing syntactic and semantic dependencies // Proceedings of the Twelfth Conference on Computational Natural Language Learning. — Association for Computational Linguistics, 2008. — P. 188–192.

144 Lluìs X. Joint Learning of Syntactic and Semantic Dependencies : Ph.D. thesis / Xavier Lluìs ; Master Thesis, Universitat Politecnica de Catalunya (Artificial Intelligence Program), Barcelona. — 2008.

145 Lluìs X., Bott S., Màrquez L. A second-order joint Eisner model for syntactic and semantic dependency parsing // Proceedings of the Thirteenth Conference on Computational Natural Language Learning: Shared Task. — Association for Computational Linguistics, 2009. — P. 79–84.

146 A latent variable model of synchronous parsing for syntactic and semantic dependencies / James Henderson, Paola Merlo, Gabriele Musillo, Ivan Titov // Proceedings of the Twelfth Conference on Computational Natural Language Learning. — Association for Computational Linguistics, 2008. — P. 178–182.

147 A latent variable model of synchronous syntactic-semantic parsing for multiple languages / Andrea Gesmundo, James Henderson, Paola Merlo, Ivan Titov // Proceedings of the Thirteenth Conference on Computational Natural Language Learning: Shared Task. — Association for Computational Linguistics, 2009. — P. 37–42.

148 Multi-lingual joint parsing of syntactic and semantic dependencies with a latent variable model / James Henderson, Paola Merlo, Ivan Titov, Gabriele Musillo // Computational Linguistics. — 2013. — Vol. 39, no. 4.

149 Titov I., Henderson J. A latent variable model for generative dependency parsing // Proceedings of the 10th International Conference on Parsing Technologies. — Association for Computational Linguistics, 2007. — P. 144–155.

150 Mixing and blending syntactic and semantic dependencies / Yvonne Samuelsson, Oscar Täckström, Sumithra Velupillai et al. // Proceedings of the Twelfth Conference on Computational Natural Language Learning. — Association for Computational Linguistics, 2008. — P. 248–252.

151 Dai Q., Chen E., Shi L. An iterative approach for joint dependency parsing and semantic role labeling // Proceedings of the Thirteenth Conference on Computational Natural Language Learning: Shared Task. — Association for Computational Linguistics, 2009. — P. 19–24.

152 Chen E., Shi L., Hu D. Probabilistic model for syntactic and semantic dependency parsing // Proceedings of the Twelfth Conference on Computational Natural Language Learning. — Association for Computational Linguistics, 2008. — P. 263–267.

153 Sun W., Li H., Sui Z. The integration of dependency relation classification and semantic role labeling using bilayer maximum entropy Markov models // Proceedings of the Twelfth Conference on Computational Natural Language Learning. — Association for Computational Linguistics, 2008. — P. 243–247.

154 McCallum A., Freitag D., Pereira F. C. N. Maximum entropy Markov models for information extraction and segmentation // Proceedings of the Seventeenth International Conference on Machine Learning. — Vol. 17. — Morgan Kaufmann Publishers Inc., 2000. — P. 591–598.

155 Morante R., Van Asch V., Van den Bosch A. Joint memory-based learning of syntactic and semantic dependencies in multiple languages // Proceedings of the Thirteenth Conference on Computational Natural Language Learning: Shared Task. — Association for Computational Linguistics, 2009. — P. 25–30.

156 Llus X., Carreras X., Màrquez L. Joint arc-factored parsing of syntactic and semantic dependencies // Transactions of the Association for Computational Linguistics. — 2013. — Vol. 1. — P. 219–230.

157 Завьялова О. С. О принципах построения словаря глаголов для задач автоматического анализа текста // Труды ежегодной международной конференции "Диалог 2004". — 2004.

158 Падучева Е. В. Динамические модели в семантике лексики. — М.: Языки славянской культуры, 2004.

159 Апресян Ю. Д. Избранные труды. Том I. Лексическая семантика (синонимические средства языка). — М.: Языки русской культуры, 1995. — P. 472.

160 RU-EVAL-2012: Evaluating dependency parsers for Russian / Anastasia Gareyshina, Maxim Ionov, Olga Lyashevskaya et al. // Proceedings of COLING 2012: Posters. — The COLING 2012 Organizing Committee, 2012. — P. 349–360.

161 Синтаксический анализатор системы ЭТАП и его оценка с помощью глубоко размеченного корпуса русских текстов / И. М. Богуславский, Л. Л. Иомдин, Д. Р. Валеев, В. Г. Сизов // Труды международной конференции "Корпусная лингвистика – 2008". — СПб.: Санкт-Петербургский государственный университет, 2008. — С. 56–74.

162 Синтаксически размеченный корпус русского языка: инструкция пользователя [Электронный ресурс]. — 2013. — фев. — URL: http://www.ruscorpora.ru/instruction-syntax.html.

163 Buchholz S., Marsi E. CoNLL-X shared task on multilingual dependency parsing // Proceedings of the Tenth Conference on Computational Natural Language Learning. — Association for Computational Linguistics, 2006. — P. 149–164.

164 Designing and evaluating a Russian tagset / Sharoff Serge, Kopotev Mikhail, Erjavec Tomaz et al. // Proceedings of the Sixth International Conference on

Language Resources and Evaluation (LREC'08). — European Language Resources Association (ELRA), 2008.

165 MULTEXT-East morphosyntactic specifications, version 4 [Электронный ресурс]. — 2013. — дек. — URL: http://nl.ijs.si/ME/V4/msd/html/msd-ru.html.

166 Сокирко А. В. Морфологические модули на сайте www.aot.ru // Труды международной конференции "Диалог 2004". — 2004. — P. 559–565.

167 Сокирко А. В., Толдова С. Ю. Сравнение эффективности двух методик снятия лексической и морфологической неоднозначности для русского языка (скрытая модель Маркова и синтаксический анализатор именных групп) // Интернет-математика-2005. — 2005.

168 Berg-Kirkpatrick T., Burkett D., Klein D. An empirical investigation of statistical significance in NLP // Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning. — Association for Computational Linguistics, 2012. — P. 995–1005.

169 Learning to rank for robust question answering / Arvind Agarwal, Hema Raghavan, Karthik Subbian et al. // Proceedings of the 21st ACM international conference on Information and knowledge management. — 2012. — P. 833–842.

170 Dang H. T., Kelly D., Lin J. J. Overview of the TREC 2007 question answering track // Proceedings of The Sixteenth Text REtrieval Conference, TREC 2007. — 2007.

171 Voorhees E. M., Tice D. M. The TREC-8 question answering track report // Proceedings of the Second International Conference on Language Resources and Evaluation, LREC 2000. — 2000.

172 Marchionini G. Exploratory search: from finding to understanding // Communications of the ACM. — 2006. — Vol. 49, no. 4. — P. 41–46.

173 Васильева Н. Э. Шаблоны употреблений терминов и их использование при автоматической обработке научно-технических текстов // Труды международной конференции "Диалог 2004". — 2004. — P. 96–101.

174 Большакова Е. И., Носков А. А. Программные средства анализа текстов на основе лексико-синтаксических шаблонов языка LSPL // Программные системы и инструменты: Тематический сборник. — 2010. — № 11. — С. 61–73.

175 Терминологический анализ текста на основе лексико-синтаксических шаблонов / Н. Э. Ефремова, Е. И. Большакова, А. А. Носков, В. Ю. Антонов // Труды международной конференции "Диалог 2010". — Т. 9. — М.: Изд-во РГГУ, 2010. — С. 124–129.

176 Большакова Е. И. Язык лексико-синтаксических шаблонов LSPL: опыт использования и пути развития // Программные системы и инструменты: Тематический сборник. — 2014. — № 15.

177 Russian statistical taggers and parsers [Электронный ресурс]. — 2015. — май. — URL: http://corpus.leeds.ac.uk/mocky/.

178 Золотова Г. А., Караулов Ю. Н. Синтаксический словарь: Репертуар элементарных единиц русского синтаксиса. — Наука, 1988.

ПРИЛОЖЕНИЯ

Приложение 1. Особенности реализации программного обеспечения для семантического и семантико-синтаксического анализа. Программная архитектура лингвистического процессора

Таблица – Особенности реализации программных компонент для лингвистического анализа текстов на ЕЯ и проведения экспериментальных исследований

Тип программного компонента	Язык реализации	Объем кода
Модули лингвистического анализа	C++	более 40 000 строк кода
Сценарии для проведения экспериментальных исследований	Python 2.7	более 3 000 строк кода

Программная архитектура лингвистического процессора

Введем два понятия:

- Анализатор – программный компонент, который либо преобразует тексты на естественном языке в структурированную форму, либо преобразует одно структурированное представление текста в другое.
- Языковой ресурс – это фрагмент лингвистической базы знаний о некотором ЕЯ: словарь, тезаурус, модель машинного обучения, система правил, конечный автомат и др.

Задача анализа текстов на естественных языках имеет ряд особенностей, которые находят свое отражение в требованиях к программному обеспечению для автоматической обработки текстов. Сформулируем эти требования.

1) Необходима возможность использовать языковые ресурсы, предназначенные только чтения, сразу в нескольких анализаторах. Анализаторы для своей работы, как правило требуют различного рода языковые ресурсы. Например, для морфологического анализатора обычно необходим словарь, в котором хранятся словоформы ЕЯ, а также статистическая модель для разрешения морфологической омонимии; для синтаксического анализатора на основе системы переходов необходима модель для выбора действий анализатора; для семантического анализатора, который описан в настоящей диссертации, нужен словарь, в котором содержится информация о предикатных словах и их ролевых структурах, словарь ролей для вопросительный слов и др. Как правило, эти ресурсы занимают большой объем оперативной памяти. В ходе работы анализатор обычно осуществляет только операции чтения этих ресурсов. Поэтому, чтобы не выделять память под несколько копий одних и тех же крупных ресурсов для нескольких анализаторов, в которых они требуются, архитектура лингвистического процессора должна обеспечивать возможность разделять использование одного языкового ресурса между несколькими объектами-анализаторами. Это требование особенно важно для программ, которые должны осуществлять обработку текстов в несколько потоков.
2) Необходимо обеспечить реентерабельность анализаторов и потокобезопасность доступа к языковым ресурсам. Эти свойства требуются для создания многопоточных программных систем. Свойство реентерабельности анализаторов необходимо для создания несколько копий анализаторов в памяти, которые могут обрабатывать текст параллельно в нескольких потоках. Свойство потокобезопасности доступа к языковым ресурсам, необходимо для параллельного чтения

этих ресурсов несколькими анализаторами, работающими в разных потоках.

3) Необходимо предусмотреть возможность обрабатывать тексты на разных ЕЯ, используя одинаковые алгоритмы, но разные языковые ресурсы. Необходимо также предоставить возможность учитывать особенности разных языков в этих алгоритмах, не допуская при этом дублирования программного кода. Часто одни и те же методы и подходы можно использовать для обработки текстов на разных языках. При этом отличаются лишь небольшие элементы алгоритмов, а также языковые ресурсы, необходимые для их работы.
4) Необходима возможность гибкого конфигурирования анализаторов на этапе создания. Анализаторы для обработки текстов на естественном языке из-за своей сложности, как правило, обладают большим числом параметров. Архитектура программного обеспечения для обработки тексов на ЕЯ должна обеспечивать единообразный подход для задания этих параметров на этапе создания объектов-анализаторов.
5) Необходимо предусмотреть возможность выбирать режим работы анализатора при каждом запуске функции обработки текста, без необходимости пересоздавать анализатор заново. Помимо параметров, которые задаются на этапе создания анализатора, часто возникает необходимость переключаться между его режимами работы. Это нужно, например, когда лингвистический процессор работает как сервис в распределенной системе обработки информации, и результаты его работы используются в различных приложениях, которым требуется несколько разные структуры данных или разная лингвистическая информация.
6) Необходимо обеспечить простую интеграцию нескольких отдельных анализаторов в едином программном модуле. Для решения прикладных

задач обработки текстов на ЕЯ высокого уровня часто требуется решить множество низкоуровневых подзадач их предобработки и провести множество видов анализа для получения достаточно глубокого структурированного представления текста. Для решения каждой отдельной прикладной задачи могут потребоваться разные виды анализа и их комбинации. Это ведет к необходимости иметь множество отдельных анализаторов, каждый из которых решает свою отдельную небольшую подзадачу, при этом каждый отдельный анализатор должен иметь возможность легко интегрироваться в систему обработки текста, решающую общую прикладную задачу.

7) Необходимо обеспечить возможность отдельных анализаторов обмениваться между собой результатами обработки. Каждый анализатор имеет набор необходимых для работы входных данных и производит множество выходных. Входными параметрами могут быть, например, текст в виде строки, список токенов или предложений, морфологическая разметка, и др.; результатами работы – синтаксическая, семантическая разметка, извлеченные из текста именованные сущности и др. Необходимо иметь возможность хранения разнородной информации о тексте, но при этом также иметь возможность передать в качестве входных параметров другому анализатору некоторый элементарный срез этой информации. Анализаторы в свою очередь должны принимать на вход только тот элементарный срез информации о тексте, который непосредственно необходим для их работы.

Для выполнения этих требований лингвистический процессор можно построить на основе одной из известных программных платформ, специализирующихся на анализе текстов, таких как GATE[10], Apache UIMA[11],

[10] https://gate.ac.uk/
[11] https://uima.apache.org/

Ellogon[12] и др. Эти платформы помимо рассмотренных проблем, решают много и других задач, например, предоставляют программные компоненты для запуска анализаторов в распределенной вычислительной среде, графические инструменты для разработки и тестирования лингвистических анализаторов. Однако существует ряд причин, по которым применение подобных платформ либо невозможно, либо избыточно и ведет к излишней сложности разработки, либо просто нежелательно. Среди этих причин может быть плохая совместимость технологий платформы и технологий, в рамках которых ведется основная разработка программного обеспечения; трудоемкость создания дополнительных оберток, необходимых для интеграции разработанных анализаторов в платформу; необходимость возможности быстро и гибко вносить изменения при разработке анализатора, например, в ходе проведении научных исследований; необходимость разработки анализатора сразу под несколько платформ анализа текста. Таким образом, часто требуется не всеобъемлющая платформа, а лишь архитектура, задающая основные принципы построения лингвистического процессора, которая позволит соблюсти все указанные выше требования. В ходе исследований была разработана подобная архитектура, которая также может рассматриваться как некий паттерн проектирования лингвистического процессора.

Введем следующие понятия:

- Анализатор – объект, который решает элементарную низкоуровненвую задачу обработки текста. В нем инкапсулированы алгоритмы обработки текста;
- Языковой ресурс – объект, который предоставляет доступ к языковым ресурсам, таким как словари, модели машинного обучения и др.

[12] http://www.ellogon.org/index.php/what-is-ellogon

- Конвейер – объект, который формирует общее структурированное представление текста, необходимое для дальнейшего решения некоторой прикладной задачи высокого уровня.

Рассмотрим каждый пункт требований.

1) Возможность использования языковых ресурсов, предназначенных только чтения, сразу в нескольких анализаторах предполагает, что классы-анализаторы не должны владеть этими ресурсами и не должны контролировать время существования ресурсов. Языковой ресурс должен создаваться отдельно, время его жизни должен контролировать объект, обладающий классом более высокого уровня, например, конвейер. Сами анализаторы должны хранить ссылки на ресурсы, полученные при инициализации. Анализаторы должны быть легковесными объектами, допускающими копирование. Последнее свойство может быть использовано при разработке многопоточных приложений: инициализированный объект-анализатор может быть скопирован для работы в разных потоках, как это делается в некоторых реализациях паттерна «фабрика».
2) Реентерабельности классов-анализаторов означает, что анализаторы не должны хранить промежуточное состояние между обращениями к ним в глобальных или статических объектах. Чтобы обеспечить потокобезопасноть классов, отвечающих за доступ к лингвистическим ресурсам, необходимо запретить изменение этих объектов во времени после инициализации.
3) Чтобы обрабатывать тексты на разных ЕЯ с помощью единых алгоритмов, необходимо при реализации как можно меньше привязывать алгоритм к конкретному ЕЯ, оформляя все специфичные для языка компоненты в виде лингвистических ресурсов. Языковые ресурсы могут иметь общий абстрактный интерфейс, но специфичную для каждого

языка реализацию. Специфичные для ЕЯ элементы алгоритма можно инкапсулировать в виде виртуальных функций. Базовый абстрактный класс реализует алгоритмическое ядро анализатора. Для каждого ЕЯ может быть создан отдельный класс, унаследованный от базового абстрактного класса, в котором реализованы виртуальный функции, специфичные для заданного языка.

4) Для конфигурирования анализаторов на этапе создания и инициализации, предлагается передавать либо в конструктор, либо в функцию initialize указатели на объекты, управляющие языковыми ресурсами, и объект подкласса Configuration, который содержит остальные параметры анализатора.
5) Для выбора режима работы анализатора при запуске функции обработки текста предлагается помимо других параметров использовать объект подкласса Options. В этом объекте должны сохранятся все параметры, указывающие на режим работы анализатора.
6) Интеграцию нескольких анализаторов предлагается осуществлять в конвейере. Этот конвейер должен обеспечивать создание и инициализацию необходимых для его работы анализаторов, а также создание и инициализацию языковых ресурсов необходимых для их работы. Конвейер управляет временем жизни объектов-анализаторов и объектов, отвечающих за доступ к языковым ресурсам. Он также управляет потоками данных и обеспечивает выполнение анализаторов в правильном порядке. Для каждой прикладной высокоуровневой задачи обработки текстов на естественном языке (или группы задач) рекомендуется создавать отдельный класс конвейер, который интегрирует в себе все необходимые анализаторы и ресурсы для ее решения.

7) Структуры данных, которыми обмениваются анализаторы, должны представлять собой набор слоев, каждый из которых создается отдельным анализатором и может быть передан другому анализатору в качестве входного параметра отдельно ото всех остальных. Например, при проведении морфологического, синтаксического и семантического анализа, стоит предусмотреть как минимум три класса, обеспечивающих хранение результатов каждого анализа соответственно. Отдельно стоит отметить паттерн аннотаций, который предложен был в рамках проекта TIPSTER[13]. Аннотации являются универсальной графовой моделью представления лингвистической информации. В современных реализациях платформ обработки текстовой информации аннотации представляют собой список отрезков текста, с каждым из которых связана некоторая информация. Это может быть строка, число, ссылка на другую аннотацию. При использовании этой модели каждый анализатор в конвейере должен создавать новый список аннотаций, который затем может быть использован следующим анализатором более высокого уровня.

На UML-диаграмме классов ниже схематично проиллюстрирована архитектура лингвистического процессора. Методы и члены класса описаны на языке C++. ProcessorTypeBase1 – обозначает один из базовых абстрактных классов анализаторов; ProcessorTypeLanguageSpec1 – обозначает класс-анализатор, реализующий специфичные для конкретного ЕЯ элементы алгоритма; PipelineType – обозначает класс-конвейер; LangIndepResource – обозначает класс, представляющий доступ к языковому ресурсу, реализация, которого не зависит от ЕЯ; LangDepResource1 – обозначает класс, предоставляющий доступ к языковому ресурсу, реализация которого зависит от ЕЯ.

[13] www.itl.nist.gov/iaui/894.02/related_projects/tipster/docs/arch31.doc

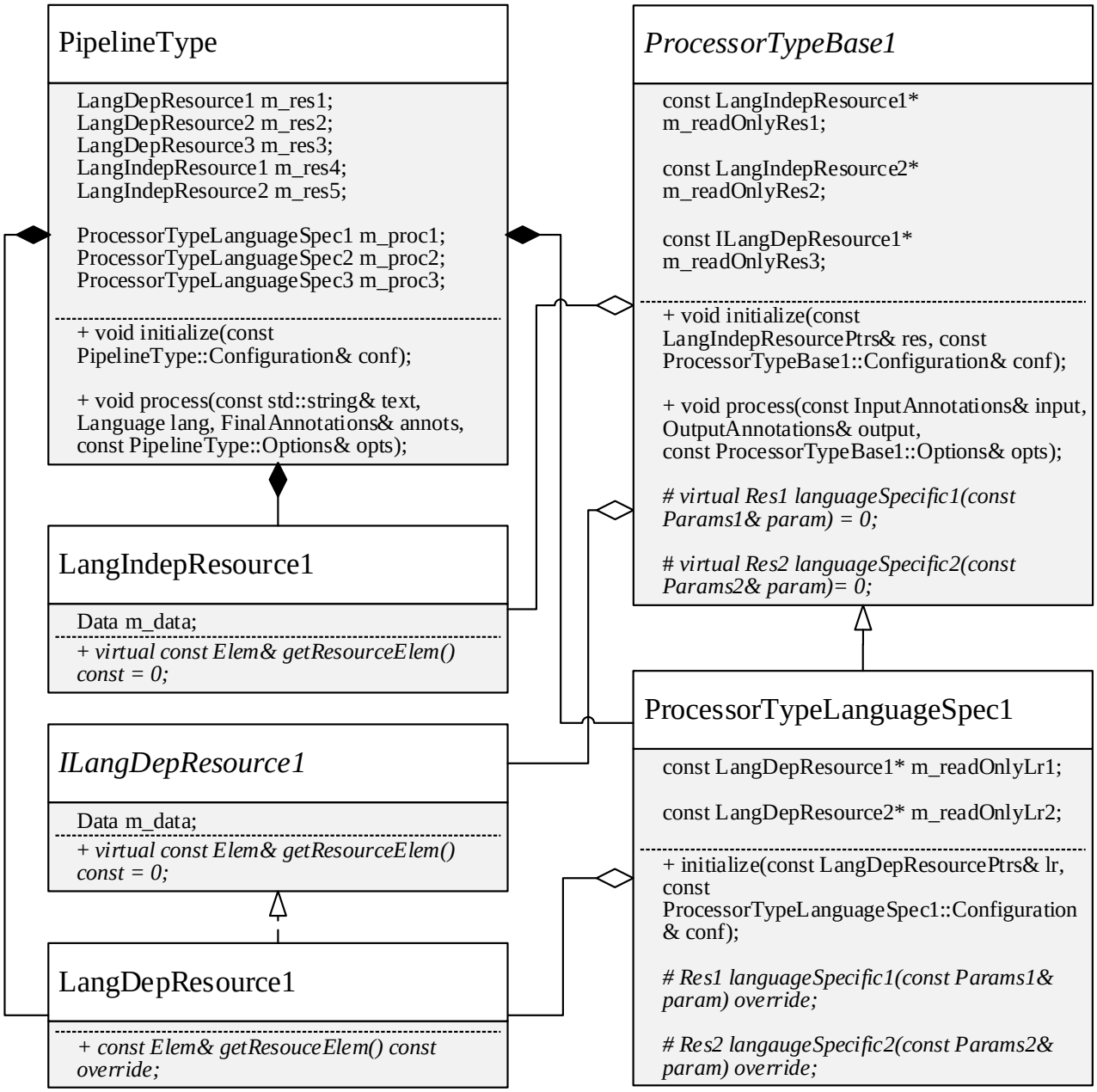


Рисунок – Архитектура лингвистического процессора

Приложение 2. Исследование эффективности синтаксического анализатора на основе MaltParser

В этом приложении описаны эксперименты по обучению синтаксического анализатора MaltParser на корпусе СинТагРус при использовании различных признаков.

Версия корпуса СинТагРус, которая использовалась в экспериментах содержит 53 439 предложений и 774 373 токенов без учета пунктуации. Корпус был преобразован в формат CoNLL Shared Task 2006/2007 [163], а морфологические характеристики токенов были преобразованы в формат, соответствующий стандарту MULTEXT East Morphosyntactic Specifications, Version 4 (MTE) [165].

Предварительные эксперименты показали, что производительность и требования к памяти многократно различаются в случаях, когда анализатор обучен на корпусе с полным набором меток (типов) синтаксических связей и на корпусе без меток, в пользу последнего. Обучение на корпусе с полным набором меток позволяет анализатору не только строить синтаксическое дерево, но и помечать связи этими метками, определяя таким образом их тип, что может оказаться полезным при решении некоторых задач. Однако часто необходимо построить только синтаксическое дерево. В этих случаях большое преимущество в скорости анализатора, обученного на корпусе без меток, делает его применение выгодным. Поэтому большая часть меток из корпуса была удалена. Оставлена небольшая часть меток, обозначающих виды сочинительных связей, метка, указывающая на то, что заданная вершина связана с виртуальной корневой вершиной «ROOT» и метка по умолчанию. Это позволяет многократно сократить требования к оперативной памяти как при обучении, так и при анализе, а также повысить скорость синтаксического анализа. К примеру, при обучении на корпусе размером 30 000 предложений (около 430 000 токенов) требуется около

14 Гб ОЗУ для корпуса с полным набором меток, в то время для обучения на преобразованном корпусе требуется всего около 2 Гб ОЗУ. В первом случае анализ 5 000 предложений (около 70 000 токенов) выполняется за 241 сек, при занятых 700 Мб ОЗУ, тогда как во втором случае выполняется всего за 13 сек, при занятых 350 Мб ОЗУ.

В проведенных экспериментах в качестве базовой конфигурации анализатора и базового набора признаков использовалась конфигурация и набор признаков из работы [56], которая открыто доступна на электронном ресурсе [177]. В ней используется алгоритм разбора «nivreeager». В базовую конфигурацию была добавлена опция «covered_root = left», которая указывает анализатору провести предварительную обработку пунктуации в корпусе, задействовав механизм псевдо-проективного анализа. Поскольку пунктуация в СинТагРус не соединяется связями с другими токенами, в представлении MaltParser она привязывается к вспомогательной вершине – корню синтаксического дерева. Это приводит к появлению большого количества непроективных связей, соединяющих корень дерева и пунктуацию. Большое количество непроективных связей сильно ухудшает качество синтаксического анализа. Без добавления указанной опции оценки качества анализа до 7% ниже, чем результаты, представленные в [56]. Это может объясняться тем, что в этой работе пунктуация дополнительным образом преобразовывалась. В дальнейшем базовой конфигурацией считается конфигурация из [56], в которую добавлена указанная опция.

Базовый набор признаков содержит информацию о части речи слов (POSTAG), форме слов в тексте (FORM), их лемме (LEMM) и морфологических характеристиках (FEATS), а также о метках уже установленных синтаксических связей (DEPREL). Заметим, что в базовой конфигурации весь набор морфологических характеристик рассматривается как атомарная строка.

Были проведены эксперименты со следующими наборами признаков:

- Base = Базовый = FORM + LEMMA + POSTAG + DEPREL + FEATS.
- NoForm = Базовый – FORM = LEMMA + POSTAG + DEPREL + FEATS.
- NoForm_NoLemma = Базовый – FORM – LEMMA = POSTAG + DEPREL + FEATS.
- NoForm_NoLemma_CSC = Базовый – FORM – LEMMA + КСК = POSTAG + DEPREL + FEATS + КСК.
- Base_CSC = Базовый + КСК = FORM + LEMMA + POSTAG + DEPREL + FEATS +КСК.
- NoFeats = Базовый – FEATS = FORM + LEMMA + POSTAG + DEPREL.
- NoFeats_SplFeats = Базовый – FEATS + SPLFEATS (число, падеж, род) = FORM + LEMMA + POSTAG + DEPREL + SPLFEATS. В этом наборе признаков из всех морфологических характеристик слов были оставлены лишь число, падеж и род, эти характеристики обозначены как SPLFEATS. Они использовались как отдельно, так и объединенные в одну строку.
- Base_SplFeats = Базовый + SPLFEATS = FORM + LEMMA + POSTAG + DEPREL + FEATS + SPLFEATS.

В задаче синтаксического анализа количество признаков измеряется сотнями тысяч. Поэтому для построения модели использовался один из самых быстрых методов машинного обучения – метод опорных векторов без ядра (с линейным ядром), реализованный в программном пакете LIBLINEAR[14]

Качество синтаксического анализа оценивалось с помощью UAS (unlabeled attachment score) – точность установления управляющего токена.

[14] http://www.csie.ntu.edu.tw/~cjlin/liblinear/

В качестве обучающего корпуса использовался подкорпус СинТагРус, содержащий около 40 000 предложений и около 580 000 токенов без учета пунктуации, подкорпус для тестирования содержит 5 000 предложений и около 70 000 токенов без учета пунктуации.

В таблице ниже представлены значения UAS, рассчитанные при тестировании синтаксического анализатора, обученного на разных наборах признаков. Все признаки за исключением КСК являются эталонными. Все оценки получены без учета пунктуации.

Таблица – Точность синтаксического анализа

Набор признаков	**UAS**
Base	86,5
NoForm	86,3
NoForm_NoLemma	80,4
NoForm_NoLemma_CSC	81,3
Base_CSC	86,6
NoFeats	84,1
NoFeats_SplFeats	86,4
Base_SplFeats	86,7

Лучшие результаты показывают анализаторы, обученные на базовом наборе признаков (Base) и его модификациях. Однако такая модель сильно привязана к лексике и словоупотреблениям, присутствующим в обучающем корпусе. Анализатор, обученный на наборе признаков, в котором не учитываются формы слов (NoForm) показывает незначительное уменьшение UAS (на 0,2%), однако скорость обучения и анализа у него существенно выше, а требования к памяти ниже. Полное удаление признаков, учитывающих лексику (NoForm_NoLemma), сильно снижает точность анализа до 80,4%. При этом добавление КСК существительных (NoForm_NoLemma_CSC) в качестве признака хотя и дает небольшой прирост точности (на 0,7%), остается существенный разрыв между

анализаторами с учетом лексики и без. Морфологические характеристики слов FEATS достаточно сильно влияют на качество анализа, если их не учитывать (NoFeats), UAS теряет 2,4%. Полная строка морфологических характеристик может быть заменена набором из трех характеристик: число, падеж, род (NoFeats_SplFeats), при незначительном снижении точности на 0,1%.

Была построена зависимость оценки точности синтаксического анализа от размера корпуса. При этом использовалась конфигурация «Base». Корпус, был поделен на тестовую и обучающую части следующим образом: тестовая содержит 5 439 предложений и 78 396 токенов (без учета пунктуации), часть для обучения, содержит максимум 48 000 предложений. Обучающий корпус наращивался с 5 000 предложений с шагом по 5 000 предложений. На рисунке ниже представлена зависимость UAS анализатора от размера обучающего корпуса.

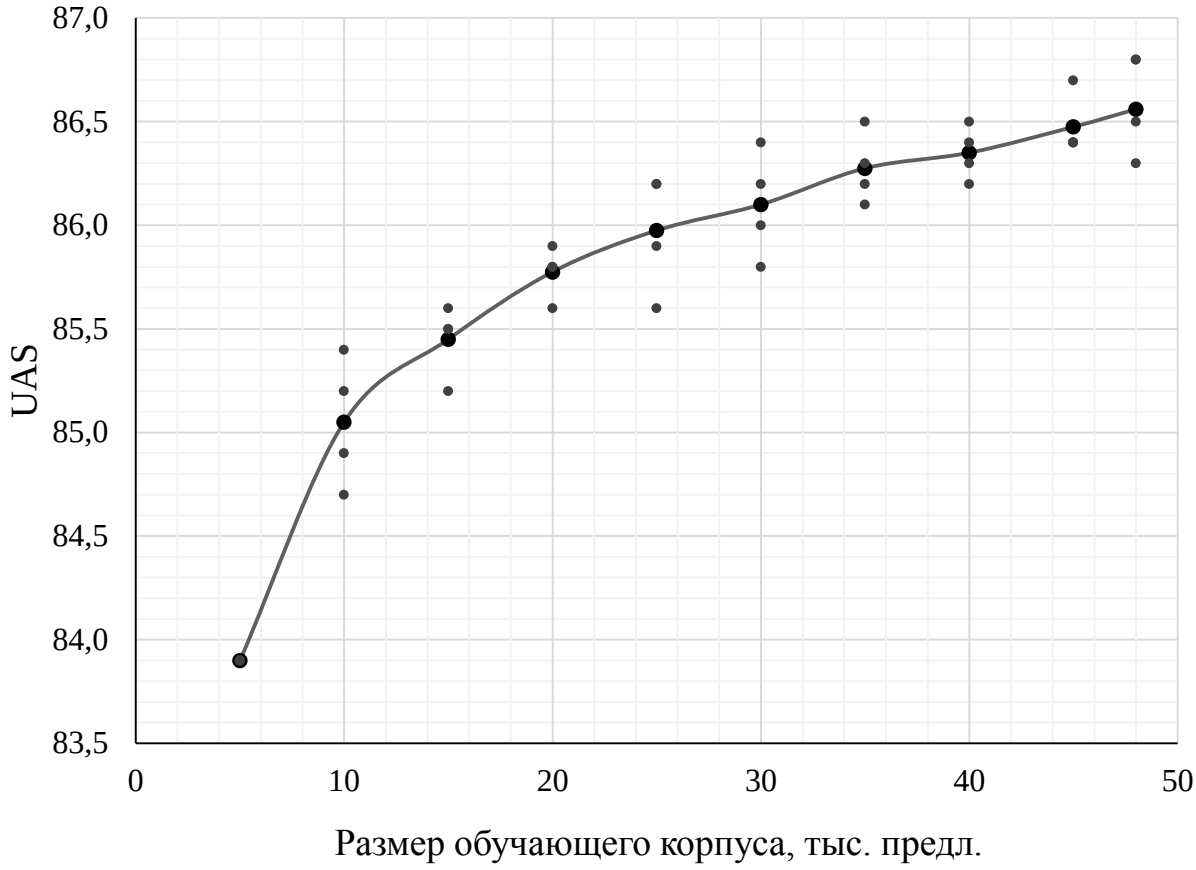


Рисунок – Зависимость точности синтаксического анализа от размера обучающего корпуса

Эксперимент показывает, что даже сравнительно небольшие корпуса от 10 000 предложений позволяют обучать синтаксические анализаторы, обладающие приемлемой точностью для решения многих прикладных задач.

Приложение 3. Описание семантического словаря

Семантический словарь состоит из трех разделов:

- раздел ролевых структур предикатных слов;
- раздел периферийных ролей;
- раздел вопросительных слов.

Раздел ролевых структур предикатных слов представляет собой список словарных статей. Словарные статьи содержат предикатные слова, обозначающие одну и ту же ситуацию, а также их ролевые структуры. Каждая словарная статья содержит:

- список лемм предикатных слов, с помощью которых ситуация может быть обозначена в тексте, для глаголов также указывается их класс по классификации Г.А. Золотовой [178];
- список описаний центральных (обязательных) семантических ролей, которые могут играть участники ситуации;
- комментарии и неразмеченные примеры, характеризующие ситуацию, которая обозначается предикатными словами из словарной статьи.

Каждое описание семантической роли состоит из:

- идентификатора / названия роли;
- множества категориально-семантических классов семантических аргументов для заданной роли;
- списка групп грамматических признаков.

Категориально-семантический класс (КСК) существительных – это обобщенное значение, которое характеризует слова, принадлежащие к одной из следующих категорий:

- «предметное» – материальная сущность;
- «признаковое» – нематериальная сущность;
- «личное» – лицо, способное целенаправленно действовать, мыслить;

- «пространственное» – место в пространстве;
- «темпоративное» – момент времени;
- «единица_измерения» – имя со значением единицы измерения;
- «параметр_измерения» – имя со значением параметра измерения;
- «количественное» – имя со значением количества.

Группа грамматических признаков определяет морфологическую реализацию семантической роли в тексте. В группе два признака: падеж и предлог (предлог может состоять из нескольких слов) семантического аргумента. Как правило, для каждой семантической роли в словарной статье указана не одна, а несколько групп таких признаков.

В разделе ролевых структур предикатных слов для русского языка содержится 2 634 словарных статей. Пример словарной статьи приведен в таблице ниже.

Таблица – Пример словарной статьи из раздела ролевых структур предикатных слов

<table>
<tr><td>Леммы предикатных слов</td><td colspan="3">Отправить, отправлять, направить, направлять, послать посылать, сослать, выслать, слать (класс: акциональные).</td></tr>
<tr><td colspan="4">Семантические роли</td></tr>
<tr><td rowspan="7">Аблатив
(Исходная точка действия)</td><td>КСК</td><td colspan="2">Морфологические признаки</td></tr>
<tr><td rowspan="6">Локативное
Предметное</td><td>Предлог</td><td>Падеж</td></tr>
<tr><td>Из</td><td>Родительный</td></tr>
<tr><td>Из-за</td><td>Родительный</td></tr>
<tr><td>Из-под</td><td>Родительный</td></tr>
<tr><td>От</td><td>Родительный</td></tr>
<tr><td>С</td><td>Родительный</td></tr>
</table>

Адресат (лицо или предмет, к которому обращено информативное, донативное или эмотивное действие)	**КСК**	**Морфологические признаки**	
	Личное	**Предлог**	**Падеж**
		К	Дательный
Дестинатив (назначение действия или предмета)	**КСК**	**Морфологические признаки**	
	Любой	**Предлог**	**Падеж**
		Для	Родительный
Директив (направление движения, ориентированного действия или положения предмета)	**КСК**	**Морфологические признаки**	
	Локатив Предметное	**Предлог**	**Падеж**
		В	Винительный
		За	Винительный
		На	Винительный
Медиатив (способ, средство действия)	**КСК**	**Морфологические признаки**	
	Предметное	**Предлог**	**Падеж**
		По	Дательный
Объект (компонент, подвергающийся действию или каузативному воздействию)	**КСК**	**Морфологические признаки**	
	Любой	**Предлог**	**Падеж**
		-	Винительный
Объектив (компонент со значением цели)	**КСК**	**Морфологические признаки**	
	Любой	**Предлог**	**Падеж**
		За	Творительный
		На	Винительный
Субъект (инициатор действия)	**КСК**	**Морфологические признаки**	
	Личное	**Предлог**	**Падеж**
		-	Именительный

В разделе периферийных ролей содержатся типичные признаки для «необязательных» семантических ролей, которые могут назначаться аргументам

ПС, но не входят в основу их ролевой структуры. Среди этих ролей – «локатив» (место действия), «темпоратив» (время действия), «инструментатив» (инструмент действия), «каузатив» (причина действия) и др. Раздел представляет собой список словарных статей, которые содержат:

- название/идентификатор семантической роли;
- множество категориально семантических классов заданной роли;
- список групп морфологических признаков, группа также содержит два признака: падеж и предлог.

Пример словарных статей из раздела периферийных ролей представлен в таблице ниже.

Таблица – Пример словарных статей из раздела периферийных ролей семантического словаря

Семантическая роль	**КСК**	**Морфологические признаки**	
		Предлог	**Падеж**
Локатив (Место действия)	Предметное Локативное	В	Предложный
		На	Предложный
		…	…
		Между	Творительный
		Над	Творительный
		…	…
Каузатив (Причина действия)	Признаковое	За	Винительный
		Благодаря	Дательный
		По	Дательный
		В силу	Родительный
		…	…

В разделе вопросительных слов содержится информация о том, какие роли могут играть вопросительные слова. Эта информация используется для определения семантических ролей наречий, союзов и некоторых местоимений, а также ролей вопросительных слов при обработке вопросительных предложений, что особенно важно в вопросно-ответном поиске.

Раздел вопросительных слов представляет собой список словарных статей, каждая из которых содержит:

- лемму вопросительных слов;
- список описаний семантических ролей, которыми могут обладать вопросительные слова.

Описание семантической роли содержит:

- название/идентификатор роли;
- список групп морфологических признаков.

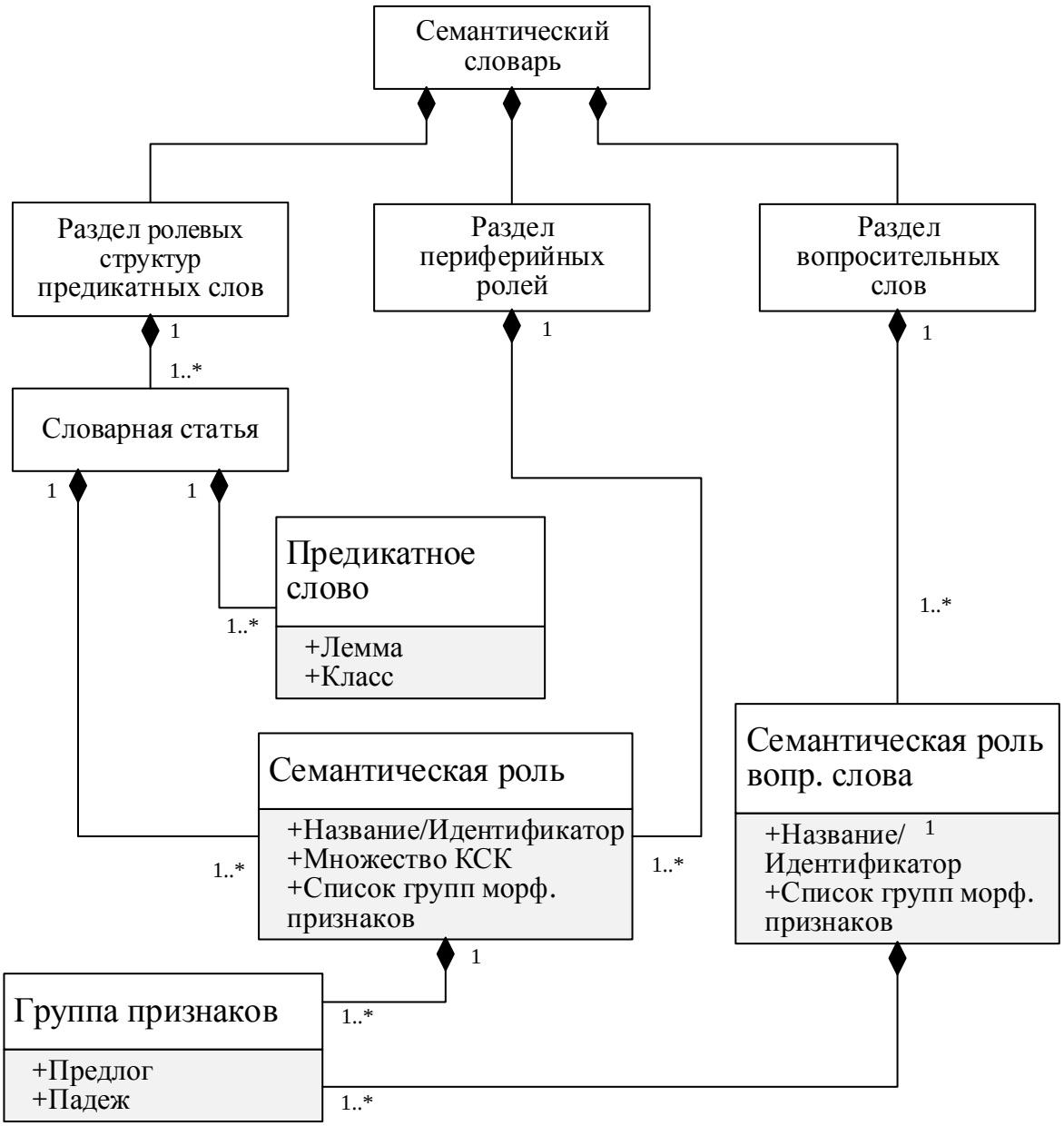
Группа морфологических признаков также, как и в остальных разделах состоит из падежа и предлога. Ниже в таблице приведены примеры словарных статей из раздела вопросительный сов семантического словаря.

Таблица – Примеры словарных статей раздела вопросительных слов семантического словаря

<table>
<tr><th>Вопросительное слово</th><th colspan="3">Семантические роли</th></tr>
<tr><td rowspan="6">Кто</td><td rowspan="2">Название</td><td colspan="2">Морфологические признаки</td></tr>
<tr><td>Предлог</td><td>Падеж</td></tr>
<tr><td rowspan="3">Адресат</td><td>-</td><td>Дательный</td></tr>
<tr><td>к</td><td>Дательный</td></tr>
<tr><td>…</td><td>…</td></tr>
<tr><td rowspan="3">Делибератив</td><td>-</td><td>Винительный</td></tr>
<tr><td>про</td><td>Винительный</td></tr>
<tr><td>о</td><td>Предложный</td></tr>
<tr><td>…</td><td></td><td></td></tr>
<tr><td rowspan="6">Что</td><td rowspan="2">Название</td><td colspan="2">Морфологические признаки</td></tr>
<tr><td>Предлог</td><td>Падеж</td></tr>
<tr><td rowspan="4">Инструментив</td><td>-</td><td>Творительный</td></tr>
<tr><td>В</td><td>Винительный</td></tr>
<tr><td>При помощи</td><td>Родительный</td></tr>
<tr><td>С помощью</td><td>Родительный</td></tr>
</table>

		…	…
	Каузатив	Морфологические признаки	
		Предлог	Падеж
		-	Родительный
		За	Винительный
		От	Родительный

Ниже на рисунке представлена структура семантического словаря в виде UML диаграммы.



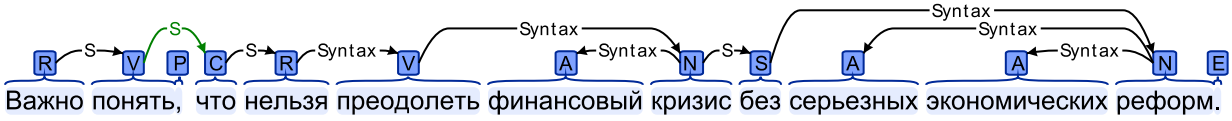
Приложение 4. Примеры, в которых разработанная интегрированная система семантико-синтаксического анализа откорректировала синтаксическую и семантическую структуру предложения

На рисунках связи в синтаксическом дереве, добавленные системой семантико-синтаксического анализа, подсвечены красным.

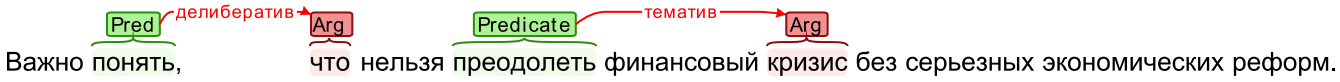
Пример 1. «Важно понять, что нельзя преодолеть финансовый кризис без серьезных экономических реформ.»

Обычный анализ:

Синтаксическое дерево:

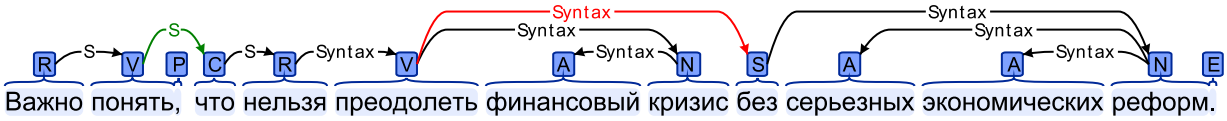


Ролевая структура предложения:

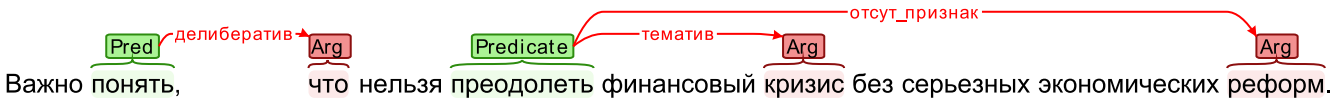


Семантико-синтаксический анализ:

Синтаксическое дерево:



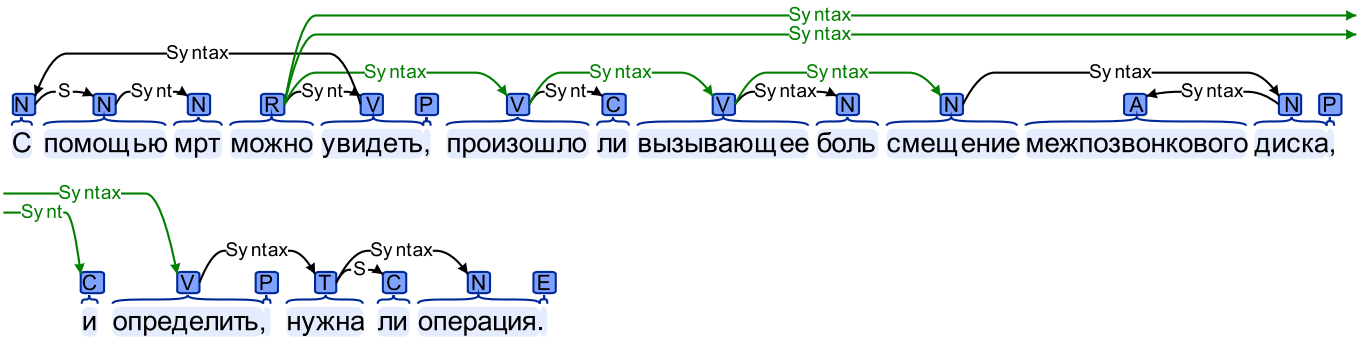
Ролевая структура предложения:



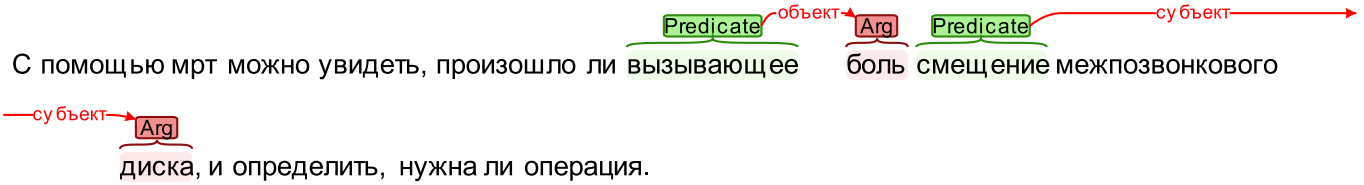
Пример 2. «С помощью мрт можно увидеть, произошло ли вызывающее боль смещение межпозвонкового диска, и определить, нужна ли операция.»

Обычный анализ:

Синтаксическое дерево:

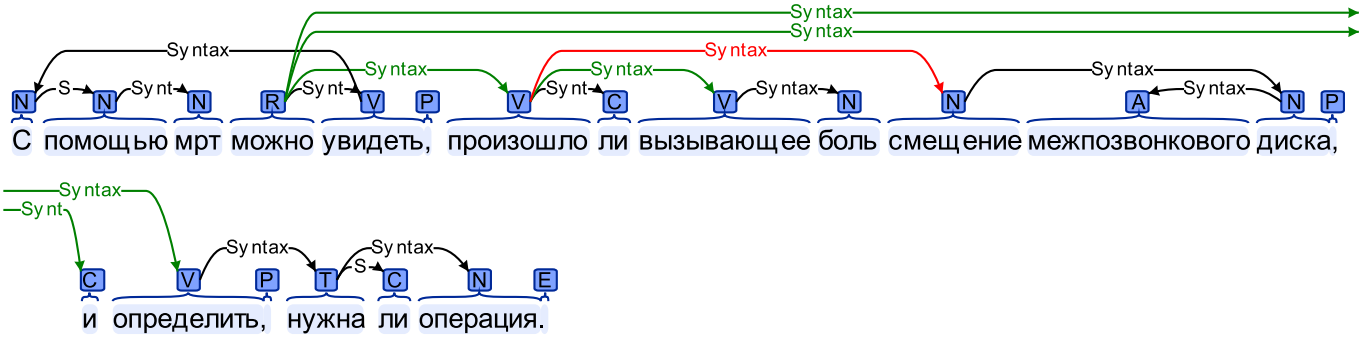


Ролевая структура предложения:

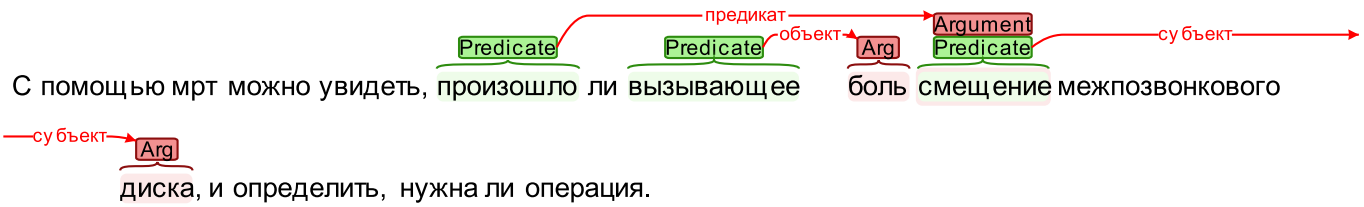


Семантико-синтаксический анализ:

Синтаксическое дерево:



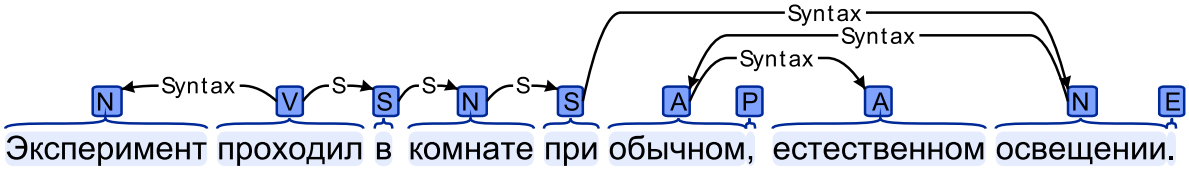
Ролевая структура предложения:



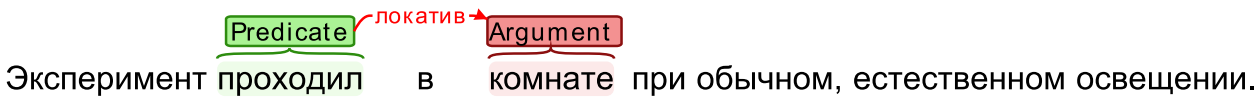
Пример 3. «Эксперимент проходил в комнате при обычном, естественном освещении.»

Обычный анализ:

Синтаксическое дерево:

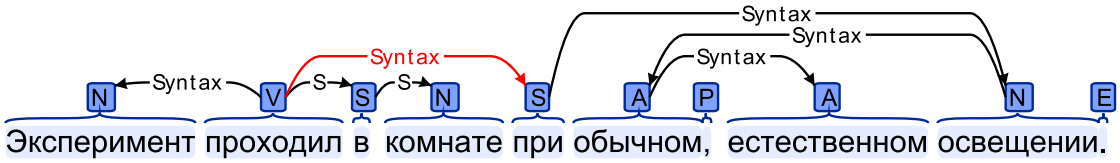


Ролевая структура предложения:

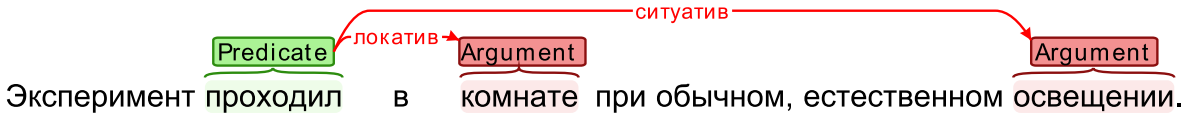


Семантико-синтаксический анализ:

Синтаксическое дерево:



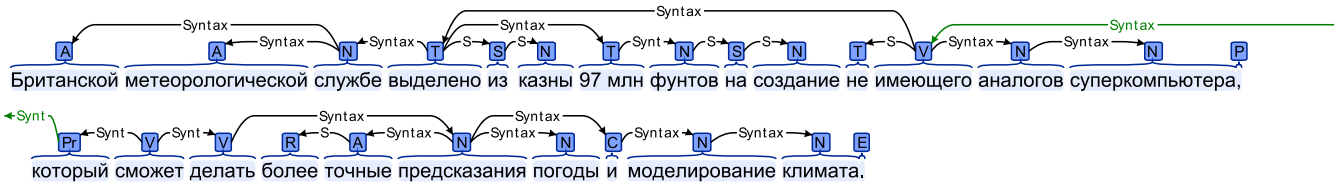
Ролевая структура предложения:



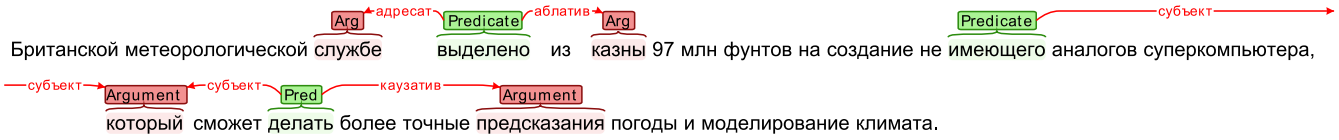
Пример 4. «Британской метеорологической службе выделено из казны 97 млн фунтов на создание не имеющего аналогов суперкомпьютера, который сможет делать более точные предсказания погоды и моделирование климата».

Обычный анализ:

Синтаксическое дерево:

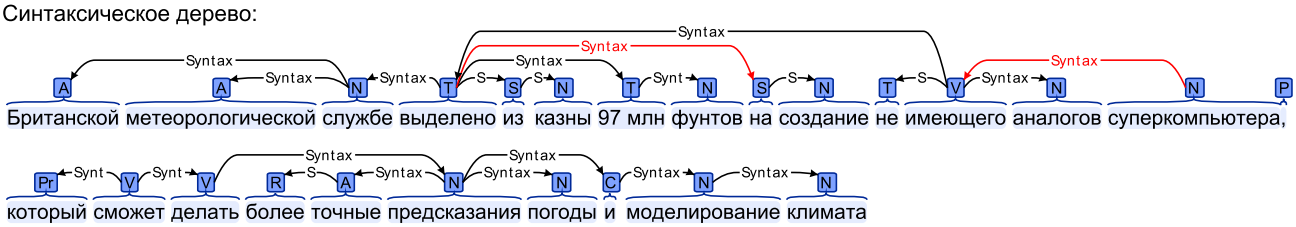


Ролевая структура предложения:

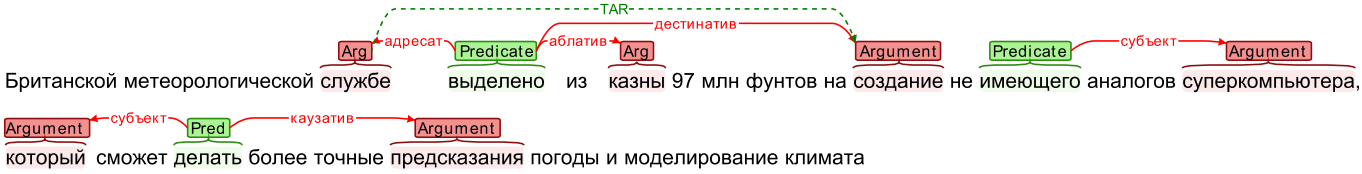


Семантико-синтаксический анализ:

Синтаксическое дерево:

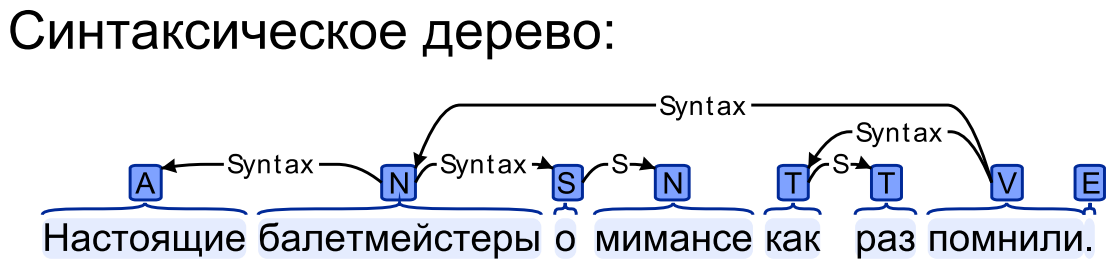


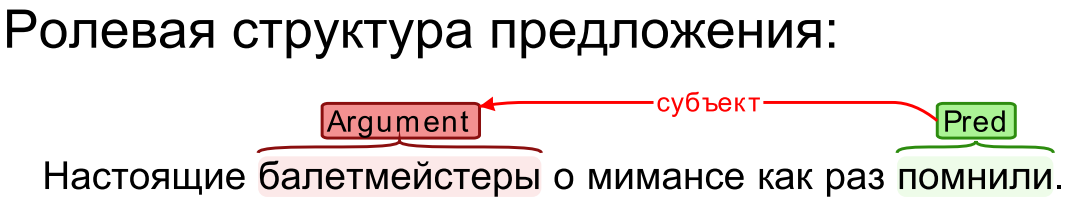
Ролевая структура предложения:



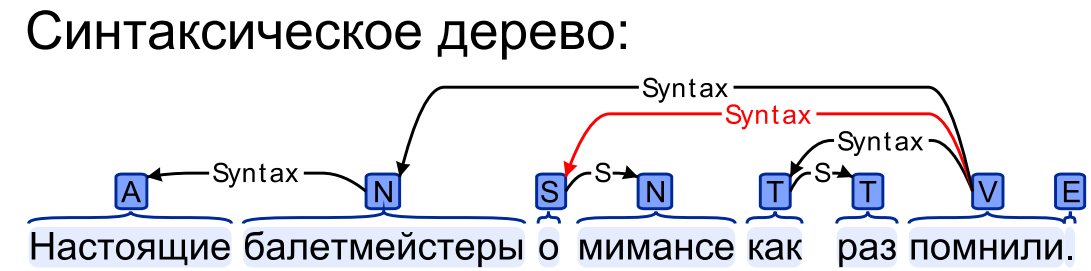
Пример 5. «Настоящие балетмейстеры о мимансе как раз помнили.»

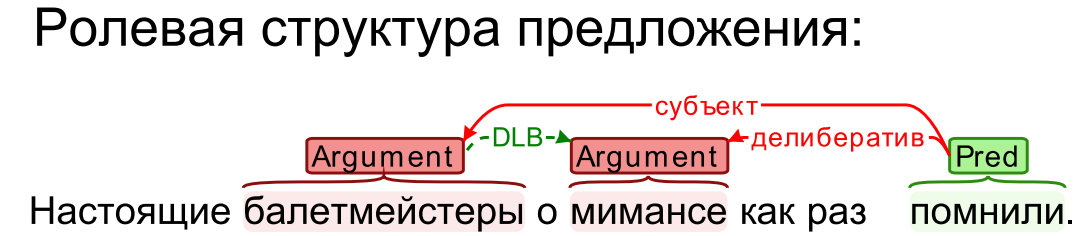
Обычный анализ:





Семантико-синтаксический анализ:

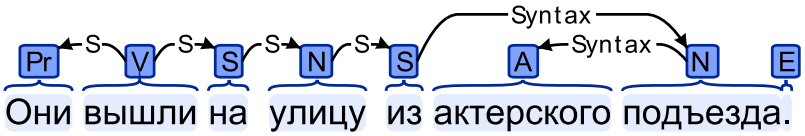




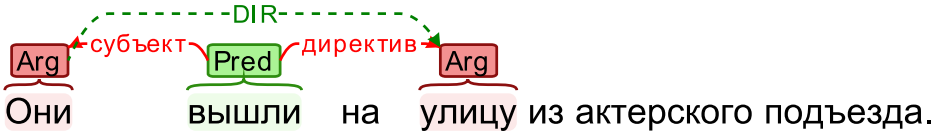
Пример 6. «Они вышли на улицу из актерского подъезда.»

Обычный анализ:

Синтаксическое дерево:

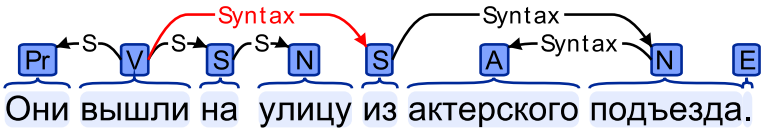


Ролевая структура предложения:

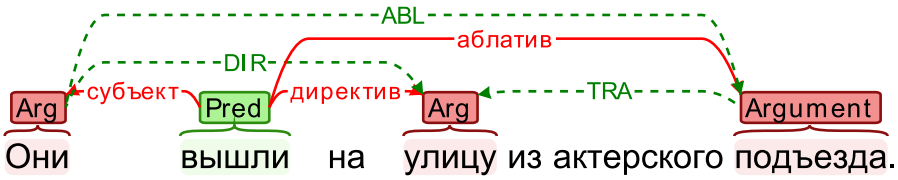


Семантико-синтаксический анализ:

Синтаксическое дерево:



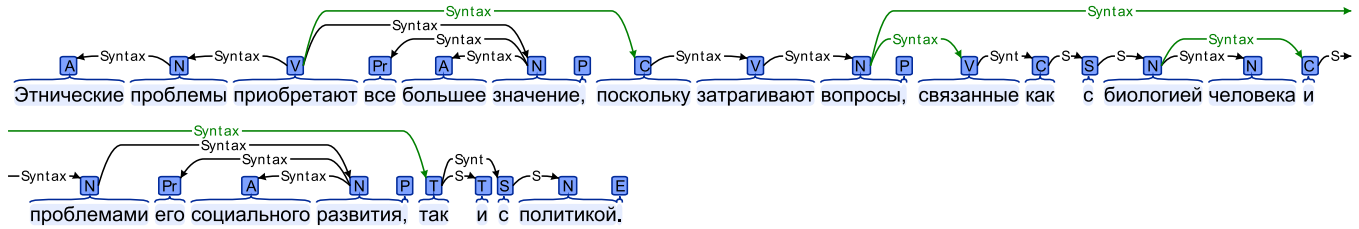
Ролевая структура предложения:



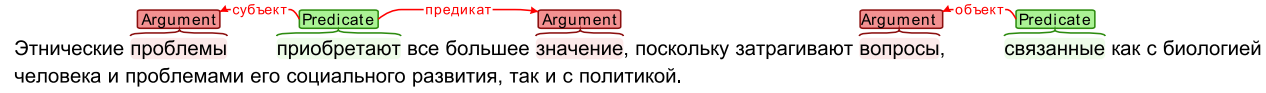
Пример 7. «Этнические проблемы приобретают все большее значение, поскольку затрагивают вопросы, связанные как с биологией человека и проблемами его социального развития, так и с политикой.»

Обычный анализ:

Синтаксическое дерево:

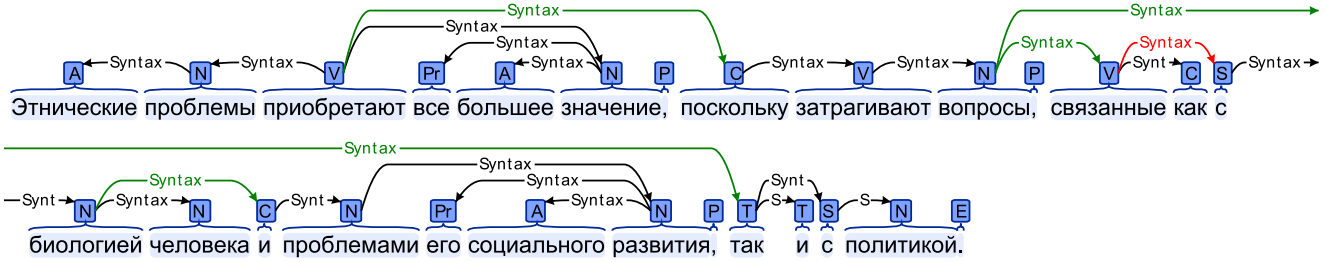
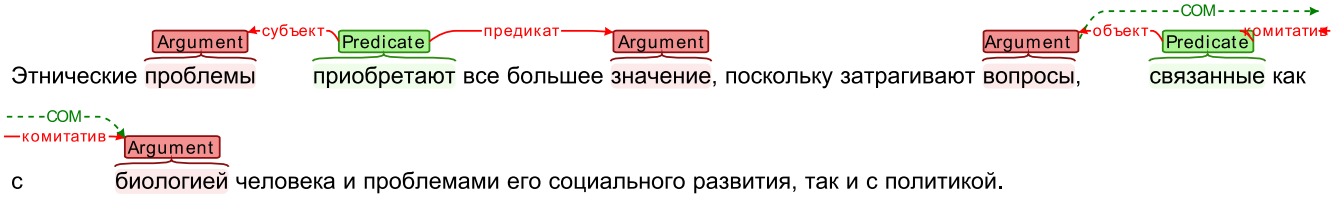


Ролевая структура предложения:



Семантико-синтаксический анализ:

Синтаксическое дерево:



Ролевая структура предложения: