КОМПЬЮТЕРНАЯ ЛИНГВИСТИКА И ОБРАБОТКА ЕСТЕСТВЕННОГО ЯЗЫКА

УДК 004.912 ISSN 1995-5499
DOI: https://doi.org/10.17308/sait.2020.1/2630
Поступила в редакцию 23.12.2019
Подписана в печать 15.03.2020

СОЗДАНИЕ ИНСТРУМЕНТА СЕМАНТИКО-СИНТАКСИЧЕСКОГО АНАЛИЗА ТЕКСТОВ НА РУССКОМ ЯЗЫКЕ

© 2020 А. С. Поречный✉

Московский авиационный институт (национальный исследовательский университет)
Волоколамское шоссе, д. 4, 125993 Москва, Российская Федерация

Аннотация. Обработку естественного языка можно разбить на несколько этапов, однако, если рассматривать их отдельно друг от друга, то возникают сложности в анализе, которые могут быть разрешены только на последующем этапе. Особенно это очевидно на синтаксическом этапе, где установка верных связей между словами зависит от «смысла» текст, т. е. от семантики. Поэтому предлагается объединить синтаксический и семантический этапы анализа текста в семантико-синтаксический. Семантико-синтаксический анализ позволяет учитывать семантику уже на уровне синтаксического анализа, что дает возможность добиться уменьшения неоднозначности в тексте и повысить качество анализа. Для реализации алгоритма предлагаемого этапа анализа выделены правила установления связей между словами, а также разработаны алгоритмы устранения неоднозначности слова и поиска связей слов в пределах опорного оборота и предложения. Также приведены результаты апробации реализованного инструмента семантико-синтаксического анализа.
Ключевые слова: обработка естественного языка, синтаксический анализ, семантический анализ. семантико-синтаксический анализ, фильтр устранения неоднозначности, опорные слова, опорные обороты.

ВВЕДЕНИЕ

Одним из направлений компьютерной лингвистики является «обработка естественного языка» (Natural Language Processing, NLP). Обработка естественного языка ставит перед собой задачи исследования и разработки методов и систем, обеспечивающих реализацию процесса общения человека с ЭВМ на естественном языке, т. е. создание естественного-языкового интерфейса, например, с помощью распознавания звучащей речи и синтеза речи по тексту, распознавания входного текста, разработки системы «вопрос-ответ» (Question Answering), извлечения фактов и знаний (Data Mining, Information Extraction/ Retrieval), автоматического машинного перевод и т. д.

Зачастую автоматический анализ текста разбивается на несколько этапов: графематический, морфологический, синтаксический, семантический [1], концептуальный [2], прагматический, а также применяются совместно с ними или же отдельно статистические методы анализа [3].

Однако, разные языки могут иметь разные морфологически характеристики, разные наборы правил словообразования, сочетаемости слов, образования предложений и т. д. [4]. Так, в японском языке порядок слов имеет четкие правила, а в русском языке таких правил нет и допускается: прямая, обратная и косвенная речь.

Поэтому некоторые алгоритмы или подходы к анализу естественного языка, могут

✉ Поречный Александр Сергеевич
e-mail: alex.porechny@mail.ru

подходить для одно языка, а для другого нет. Так, одним из популярных методов стемминга является алгоритм Портера, разработанный Мартином Портером в 1979 г. Смыслом алгоритма Потрета является поиск основы слова, т. е. стемминг. Предполагается что формо- и словообразующие суффиксы имеют конечное множество, и стемминг слова происходит без использования каких-либо баз основ. Данный метод позволил сократить словари на третью часть [5]. Примером использования такой методики является поддерживающийся до сих пор инструмент Snowball [6]. Однако, такой подход для русского языка является неудачным, т. к. в русском языке существует множество вариантов словообразования, причем некоторые из них пересекаются, а также существует большое количество исключений.

Графематический и морфологический этап анализ имеет множество реализаций для некоторых естественных языков [1]. Однако, например для русского языка, графематический анализ реализован в инструментах АОТ [7], Greeb [8], NLTK [9], Solarix [10], GATE и др. и морфологический анализ в JMorfSdk [11, 12], АОТ [7], FreeLing[13], MAnalyzer [14], RussianMorphology [15], GATE и др., т. е. первые два этапа имеют большое количество практических реализаций для русского языка.

Для синтаксического этапа анализа также существуют различные алгоритмы и подходы [1], которые для русского языка реализованы в ряде инструментов, например, АОТ [7], ABBYY Compreno [16], GATE [17] и т. д., но зачастую каждая из реализаций имеет свои ограничения и условия применения, а реализации, претендующие на семантический этап анализа, имеют их еще больше. Зачастую ограничения связанны с тем, что естественный язык сложно формализуем из-за ряда особенностей, таких как: омонимия, омоформия и т. д. на всех уровнях языка (фонемном, морфемном, лексическом, синтаксическом и т. д.), а также возникающей неоднозначности на уровне смысла, который может быть передан естественным языком [1, 4, 18]. Такие особенности естественного языка порождают многозначность или неоднозначность, для разрешения которой зачастую необходимо иметь результаты последующих этапов анализа. Например, для графематического анализа иногда необходимо знать является набор символов с «–» двумя словами или одним, а это будет известно, только после морфологического анализа, в таком случае возникает предморфологический этап анализа [4]. Аналогично, для синтаксического анализа иногда необходим результат семантического анализа [19, 20].

На данный момент существует множество подходов к семантическому анализу. Основное их отличие заключается в методах реализации смыслового компонента, способах и объемах предоставляемой базы знаний. В зависимости от полноты реализации этих параметров зависит и глубина понимания семантики текста.

Так, компонентный анализ опирается на то, что семантика может быть выражена с помощью «атомов смысла», то есть с помощью конечного неструктурированного набора семантических множеств. Данный метод, хотя и кажется логичным, является не совсем достоверным в силу того, что существует множество слов, имеющих несколько значений, при этом достаточно сложно описываемых [21].

Идентификация смысла по образцу предлагает отказаться от морфологического и синтаксического анализа, при этом необходимо накладывать ограничения на смысл слов, т. е. ориентировать слова на входной текст, что является существенным недостатком при анализе текста [22].

В интегральном подходе язык представляет собой совокупность четырех фрагментов: фрагмент знания, фрагмент языка как предмета, фрагмент национальной культуры и фрагмента социального пространства в их глобальном единстве и взаимообусловленности. Стержнем, который объединяет все эти фрагменты, является коммуникационная деятельность, т. е. в данном контексте язык [23, 24].

Таким образом, может сделать вывод, что в существующих подходах к семантическому анализу зачастую не применяется и игнорируется синтаксический этап анализа, что приводит к определенным ограничениям и недостаткам в этих подходах [20].

При этом, семантический этап анализа тесно связан с синтаксическим, поэтому следует их рассматривать совместно [2, 19, 20, 25]. Семантико-синтаксический анализ есть смысловой анализ — анализ, позволяющий выделять из предложения смысловую часть. Выделение смысловой части необходимо не только для развития средств компьютерной лингвистики, а также для повышения корректности переводов, создание методов более корректного реферирования текстов, поиска схожих по смыслу текстовых документов и т. д. Однако, на данный момент такого рода анализ остаётся до конца не реализованным.

1. ОБЗОР АЛГОРИТМОВ И ПОХОДОВ СИНТАКСИЧЕСКОГО АНАЛИЗА

Несмотря на то, что при анализе текста зачастую разделяют синтаксический и семантический анализ, эти два этапа анализа правильнее рассматривать совместно, т. к. однозначное определение структуры предложения не всегда возможно без семантики [2, 19, 20, 25]. Например, «критика ученого» — без контекста невозможно понять, что «кто-то критикует учёного» или «кого-то критикует ученый», или «посещение родственников» — «кто-то посетил родственников» или «кого-то посетили родственники», т. е. возникает синтаксическая омонимия, которую можно разрешить только, зная контекст. С другой стороны, семантика строится на основе синтаксиса. Например, «солнце село за село» — с точки зрения семантики «село» в обоих случаях может быть и глагол, и существительным, но синтаксис указывает, что после предлога всегда должно быть существительное, тем самым второе «село» становиться существительным. Поэтому объединение синтаксического и семантического анализа в единый более полный семантико-синтаксический анализ позволяет увеличить точность автоматического анализа текста.

Для отображения структуры предложений могут использоваться следующие модели: членов предложений, непосредственно составляющих или дерево зависимостей. Первые две модели являются неполными и лишь частично отражают структуру предложения, они зачастую условны и могут содержать ошибочные связи или неточности, при этом последняя модель является наиболее популярной [19].

В дереве зависимостей обычно вершиной становится сказуемое, от него идут связи к непосредственным подчинённым, а от них непосредственно к их подчинённым и так далее. Связь выражается в типе «управляющий — управляемый» (эквивалент «главный — зависимый»). Однако, такая структура не может отразить некоторые возможные предложения. В таких предложениях, как правило, содержатся эллиптические конструкции типа «моделирование классов и структуры алгоритма». Здесь слово «алгоритма» является управляемым словом для двух управляющих «классов» и «структуры». Получается, что для отображения предложения в формализованной структуре нужно применять не дерево, а сеть зависимостей [20].

Семантико-синтаксические структуры текстов обычно описываются в терминах частей речи (существительное, прилагательное и т. д.) словами и сопроводительной информацией, характеризующей эти слова (например, род, число, падеж, лицо, время и т. д.), а также отношениями между ними, выраженными с помощью одной из вышеописанных моделей.

В основу алгоритмов семантико-синтаксического анализа может быть положена гипотеза: «одинаковым последовательностям символов классов слов соответствуют одинаковые синтаксические структуры» [19, 25, 26]. Естественный язык является развивающимся, постоянно меняющимся и содержит бесконечное множество структур, но все же в большинстве случаев гипотеза верна и может применяться для решения многих семантико-синтаксических задач.

Например, основываясь на приведенной выше гипотезе, может быть частично решена проблема неоднозначности (омонимии, омографии и т. д.) слов. Например, слово «мыло» может быть и существительным, и глаголом в зависимости от окружающих слов, т. е. от структуры предложения зависит, какой сло-

воформой будет являться слово «мыло». Во фразе «мама мыла раму» не ясно слово «мыла», но благодаря схожей структуре в «мама протирала раму» (т. е. сущ.+ гл.+ сущ.), можно определить, что «мыла» является глаголом.

Расширенный вариант этого метода применяется в системе фразеологического перевода RETRANS [27, 28], где за основу берутся словари структур. Для создания таких словарей, требуется извлечь из текста множество структур. Для их получения формируются микротексты, один микротекст — это слово + 5 слов справа + 5 слов слева, предложения. Далее составляется краткое описание каждого слова (окончание, грамматический класс и т. д.). После формируется структура каждого микротекста. В итоге получаются сегменты, которые с большой вероятностью определяют набор характеристик для центрального слова. Эти сегменты и являются исходными структурами. Данная система позволяет устранить омонимию с вероятностью 0,99. Недостатком системы является наличие огромных словарей структур, которые со временем устаревают и требуют ручного контроля актуальности структур.

Помимо этого, можно применить вышеизложенную гипотезу не только в синтаксическом анализе, но и в фильтре, уменьшающем количество неоднозначных слов, который также можно применить перед статистическим или иным анализом текста.

Фильтр основан на том, что слово в предложении с однозначной словоформой или однозначной частью речи определяет множество возможных синтаксических структур (следствие из гипотезы), которые в свою очередь формируют подмножество возможных частей речи, которые могут быть употреблены с исходным словом. Далее с применением правил идет сопоставление частей речи из полученного подмножества с частями речи словоформ неоднозначного слова. В случае совпадения у неоднозначного слова точно определяется или словоформа, или ее часть речи в данном предложении. Например, «солнце село за село», слово «солнце» имеет две словоформы, но часть речи однозначно существительное, а «село» имеет три словоформы — две в роли существительного в им. или тв. падеже и одна в роли глагола. Так как по правилам существительное может управлять справа стоящим существительным, только если последнее в родительном падеже, значит первое «село» не является существительным, в тоже время оно в роли глагола согласуется со словом «солнце» по роду, следовательно, «солнце» однозначно определило словоформу первого слова «село». Аналогично однозначный предлог «за» определяет словоформу второго слова «село», т. к. по правилам после предлога должно стоять существительное, согласованное с предлогом по падежу.

Таким образом, благодаря применению фильтра может быть устранена неоднозначность слов, в первом случае — до однозначной части речи, во втором — до однозначной словоформы. Такой фильтр не требует словарей, но необходимы правила для верного сопоставления частей речи.

Алгоритм, основанный на применении такого фильтра, может быть дополнен элементами прагматического анализа, исходя из гипотезы, что если «два именных словосочетания находятся в отношении «род-вид», то с высокой вероятностью в таком же отношении находятся и их опорные слова» [19]. В результате проверки данной гипотезы было установлено, что она выдает неправильный результат не более, чем в 15 % случаев [19].

2. РАЗРАБОТКА АЛГОРИТМА ВЫДЕЛЕНИЯ СЛОВОСОЧЕТАНИЙ

2.1. Алгоритм выделения словосочетаний

Привычная модель предложения, в которой смысловой основой выступает «сказуемое + подлежащее», является частным случаем, зачастую все предложения строятся на опорных словах (существительное, глаголы и т. д.), т. е. на основе определённого «скелета» [19].

В роле опорных слов могут выступать глаголы (Г), существительные (С) и отглагольные существительные (СГ), связанные между собою беспредложным и предложным управ-

лением, а также краткие прилагательные (ГП) и краткие причастия (ГП), отглагольные наречия и деепричастия (НГ), полные причастия и отглагольные прилагательные (ПГ), безличные глаголы (БезЛГ). Остальные же части речи выступают в роли определителей перечисленных выше классов или их связок (полные прилагательные (П), наречия (Н), союзы (СЗ), предлоги (Р), частицы (Ч)). Класс глаголов включает в себя на три подкласса: личный глагол (ЛГ), глагол прошедшего времени добавляется к классу ГП, инфинитив (ГИ) [19].

Каждый класс содержит определенный набор атрибутов, атрибуты, частично взяты и переработаны из книги [19], приведены в табл. 1.

В результате исследования, описанного в [19], на основе 1160 предложений, взятых из реферативных журналов по информатике и электро-вычислительной технике, были построены деревья зависимостей. После обработки полученных данных было установлено, что в зависимости от того, какой частью речи является главное слово, расстояние между главным и зависимым словом меняется. При-

Таблица 1. Классы частей речи и их атрибуты
[Table 1. Parts of speech classes and their attributes]

№	Часть речи	Класс* (сокращение)		Атрибуты	Среднее расстояние между словами
1	Личная форма глагола	ГЛ	Г	Число, лицо, род, модель управления**	3,2 (4)
2	Глагол прошедшего времени, краткое прилагательное и причастие	ГП		Род, число, модель управления**	3,5 (4)
3	Инфинитив	ГИ		Модель управления**	2,8 (3)
4	Безличные глаголы	БезЛГ		Время, род, число, модель управления**	4
5	Существительное	С		Род, число, падеж	1,7 (2)
6	Прилагательное	П		Род, число, падеж	Не управляющий класс
7	Наречие	Н			Не управляющий класс
8	Предлог	Р		Модель управления**	1,6 (2)
9	Отглагольное существительное	СГ	*Г	Род, число, падеж, модель управления**	1,7 (2) (аналогично С)
10	Отглагольное прилагательное и причастие	ПГ		Род, число, падеж, модель управления**	1,8 (2)
11	Отглагольное наречие и деепричастие	НГ		Модель управления**	2,3 (3)
12	Прочие	ПР			Не управляющий класс
13	Местоимения	М		Род, число, падеж	Не управляющий класс

* — условное обозначение частично взято из книги [20].

** — модель управления в данной версии инструмента не реализована, но планируется в следующей версии для улучшения качества выделения словосочетаний.

чем минимальное расстояние равнялось 1, а максимальное от 13 до 44, среднее от 1,7 до 3,5 в зависимости от части речи [19]. А также оказалось, что «63% из общего числа пар связанных слов расположены контактно; 80 % либо контактно, либо разделены только одним словом; 88 % — либо контактно, либо разделены одним — двумя словами. На остальные пары слов … приходилось 12 %» [19]. То есть можно сделать вывод, что больший процент зависимых слов находится на расстоянии от 1 до 2–4 слов в зависимости от части речи главного слова. В табл. 1 приведено среднее и округленное значения (в скобках) расстояния между зависимым и главным словом.

Алгоритм выделения словосочетаний можно разбить на несколько этапов:

1. Графематический этап представляет выделение слов в предложении и определение положения разделительных знаков препинания.

2. Применение морфологического анализа каждого слова, выявление опорных слов.

3. Выделение опорных оборотов.

4. Применение фильтра для разрешения неоднозначности слов.

5. Связывание опорных и не опорных слов, связывание опорных слов между собой в пределе опорного оборота.

6. Установка связей между опорными оборотами.

Первый этап является тривиальным, включает в себя разделение текста на слова и предложения. Второй выполняется с помощью библиотеки JMorfSdk [12], использующей «Грамматический словарь русского языка» А. А. Зализняка.

На третьем этапе происходит выделение опорных оборотов. Границей опорного оборота являются запятые или другие знаки препинания, а также разделительные союзы, причем количество опорных слов обязательно больше нуля.

2.2. Установление связей между главными и зависимыми словами

После разбиения предложения на опорные обороты. В каждом из них происходит связывание опорных и не опорных слов, а также опорных слов между собой, если их было больше одного, в конечном итоге получается дерево зависимостей опорного оборота, вершиной которого является главное слово опорного оборота. Связывание происходит по правилам, описанным в табл. 2, часть правил сформулирована на основе работы Г. Г. Белоногова [19].

Вершиной дерева зависимости опорного оборота обязательно будет опорное слово, оно может:

1. Быть зависящим (управляющим, главным) или зависимым (управляемым) другого главного опорного слова.

2. Иметь общее главное или управляемое слово по отношению к окружающим опорным словам, то есть быть на одном уровне.

3. Находится в разных простых предложениях в составе сложного, тогда прямой связи между ними нет.

Идея такого алгоритма заключается в том, что в естественном языке большинство слов согласуются между собой по определенным правилам, например, с помощью совпадения рода, числа и падежа между существительным и прилагательным. Конечно, такой подход дает определенный процент ошибок в процессе установления связей между словами, ввиду сложности формализации естественного языка, особенно русский язык, и тем не менее большинство связей по таким правилам можно установить.

2.3. Алгоритм работы фильтра для разрешения неоднозначности слов

Проблема разрешения неоднозначности слов (совпадение словоформ у некоторых слов или наличие несколько одинаковых словоформ у слова) является актуальной до сих пор. Зачастую она решается с применением статистики, то есть если во время анализа встречается неоднозначное слово, идет комбинация каждого значения слова и составление статистики встречаемости в той или иной сочетаемости с соседними словами. Главный недостаток такого подхода в том, что при увеличении количества неоднозначных

Таблица 2. Правила установления связей между зависимыми и главными словами
[Table 2. Rules for establishing relationships between dependent and main words]

№	Класс главного слова	Класс зависимого слова	Правило*	Отдаленность	
				слева	справа
1	С, СГ	С, СГ	Зависимое в род. падежу или же главное в дат. или твор. падеже, а зависимое не в им. падеже	нет	2
2	Р	С	Совпадение** по падежу	нет	∞
3	СГ	Р	У зависимого нет управляющего	нет	2
4	Г, *Г,	Р		нет	∞
5	С	П	Совпадение по падежу, числу, а также по роду, если ед. число	2	2
6	ПГ	С	Совпадение** по падежу	нет	2
7	НГ	С	Совпадение** по падежу	3	3
8	Г, ПГ, НГ	С	Зависимое в им. или вт. падеже; пред зависимым словом нет Р	4	4
9	Г	С	Совпадение** по падежу	4	4
10	ГИ	ГЛ	нет (проверить на совершенство)	1	нет
11	БезЛГ	С	Зависимое не в им. падеже	4	4
12	Г, С, М	ПР	Связь заключается с правым опорным словом	от Г, С, М	нет
13	Г	С, М	В пределах опорного оборота		

* — приоритет правил является их номером в этой таблице.
** — данное совпадение происходит с моделью управляемого слова, а не с его атрибутом, но в некоторых случаях подойдет совпадение по атрибутам (например, при совпадении Р + С).
Примечание: связь неопорных слов с опорным устанавливается в пределах этих опорных слов, в случае если установить связь не удалось неопорные слова присоединяются к право стоящему опорному слову. Далее осуществляется привязка опорных слов друг к другу.

слов существенно увеличивается количество комбинаций сочетаний слов.

Например, при привычном установлении связей между словами происходит следующее: «беспроводной маршрутизатор локальной и глобальной сети» — «маршрутизатор» — (какой?) — «беспроводной», «маршрутизатор» — (чего?) — «сети», «сети» — (какой?) — «локальной» и «глобальной», то есть установление идет от главного к зависимому, т. е. осуществляется поиск сверху вниз (рис. 1).

Такой алгоритм не учитывает, что главное слово может быть неоднозначным. Предлагаемый алгоритм работы фильтра позволит определить разрешить неоднозначность слов, а также установиться связь между словами.

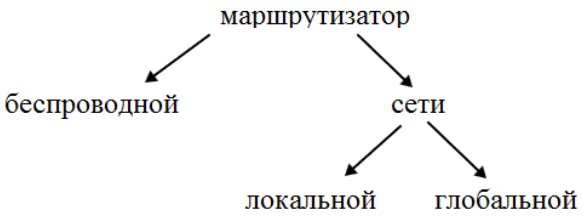

Рис. 1. Пример привычного установления связей между словами
[Fig. 1. Example of the common connection establishment between words]

Алгоритм работы фильтра для разрешения неоднозначности слов состоит из следующих этапов:

1. Нахождение однозначного слова, если найдено, переход к пункту 2, если нет, то к пункту 4.

2. Поиск главного слова к однозначному слову на основе применения описанных выше правил установления связей между главным и зависимым словами. Далее переход к пункту 3.

3. Если среди какого-либо неоднозначного слова была установлена подходящая словоформа, то устанавливается связь между ними, и т. к. главное слово стало однозначным, то применяется поиск главного слова для него и повторяется пункт 2. Если слово осталось неоднозначным, переход к пункту 1.

4. Фильтр применен.

2.4. Алгоритм установление связи между опорными оборотами

После установления взаимосвязи в пределах опорных оборотов, происходит связывание самих опорных оборотов между собой по следующему алгоритму.

Каждый оборот может быть зависимым от опорного слова других оборотов только один раз, причем чем ближе возможная связь, тем она более предпочтительная, а также зависимый оборот скорее будет стоять слева, чем справа относительного управляющего слова.

Действие 1. Сначала анализируется каждый опорный оборот на наличие подчинительного союза в начале оборота. Если такой союз удается найти, значит предложение является простым и входит в состав сложного, поэтому главное слово этого опорного оборота (слово, находящиеся в вершине графа) не может быть зависимым.

Действие 2. Далее с начала предложения берется пара опорных оборотов и происходит попытка установить связь между главными словами этих оборотов по правилам, описанным в табл. 1.2.

Действие 3. Если связь не удалось установить, происходит попытка найти управляющее слово внутри одного опорного оборота для главного слова другого оборота. Причем сначала управляющие слово ищется в левом обороте, а потом в правом.

Действие 4. Если связь была установлена, то для опорного оборота, у которого главное слово опорного оборота стало зависимым, исключается из дальнейшего поиска.

Действие 5. Действия 2–4 повторяются до тех пор, пока все обороты не будет произведено попарное сравнение всех оборотов. В итоге получается или один опорный оборот, или множество опорных оборотов, у которых главное слово будет без управляющего слова, такие опорные обороты могут считаться простыми предложениями

В результате работы алгоритма получается сеть или сети зависимостей (в частном случае дерево или деревья), в корне которых находятся сказуемое простого предложения (если сказуемое составное, то главным является одно из слов) или подлежащее (если сказуемого нет). В роли такого слова могут выступать: Г, БезЛГ, С, СГ.

3. СОЗДАНИЕ ИНСТРУМЕНТА СЕМАНТИКО-СИНТАКСИЧЕСКОГО АНАЛИЗА

Построение сетей зависимостей — важный этап в анализе предложения и текста. Проведенные исследования и анализ результатов тестирования показали, что разработанный алгоритм и реализованный на его основе инструмент позволяют выделять словосочетания из предложений на русском языке, получать результаты их статистической обработки.

Возможность выделения словосочетаний позволяет анализировать текст, быстро получать представление о смысле анализируемого текста, полностью не читая его, а, следовательно, позволяет получать не только ключевые слова, но и словосочетания на основе общего списка словосочетаний с частотами их употребления в тексте. Ключевые слова — набор слов, выделенных из текста и позволяющих определить содержание этого текста, а также выявить его тематику. Выделение ключевых словосочетаний необходимо для улучшения работы поисковых систем (анализ содержимого веб-сайта или материала для определения тематики), анализа публикаций (например, для выделения ключевых слов или составления тематических сборников, основанных не только на теме тезисов или статьи), автоматического создания аннотаций, рефератов, решения задачи классификации текстов и др.

Реализация описанных выше алгоритмов семантико-синтаксического анализа, снятия неоднозначности и набор правил для анализа были включены во фреймворк TAWT.

TAWT (Tools for Automated Work with Text) — фреймворк для автоматизированной обработки текста, который включает в себя графематический, морфологический и семантико-синтаксический этапы анализа, а также инструменты, объединяющие несколько этапов анализа и выделения словосочетаний и понятий [29, 30]. На рис. 2 представлен программный код примера вызова метода семантико-синтаксического анализа и результат его работы.

После создания инструмента семантико-синтаксического анализа была проведена оценка результатов работы реализованных алгоритмов.

Оценка качества устранения неоднозначности слов в тексте производится при помощи сравнения количества неоднозначных слов для каждой группы текстов до и после применения фильтра снятия неоднозначности, а также после установки связей между словами. Для текстов различных стилей снятие неоднозначности на тестовых данных (около 100 текстов) доходит до 50 %, причем в художественных текстах неоднозначности разрешаются лучше.

Для различных стилей текста качество установления связей разное, это обуславливается тем, что в научном (около 70 % верно установленных связей), публицистическом (около 80 %) и техническом (около 75 %) тексте имеются более простые конструкции и связанные слова стоят близко друг другу, а в художественных текстах (50–60 %) распространены сложные конструкции, поэтому связанные слова могут находится на больших расстояниях.

Скорость выполнения семантико-синтаксического анализа зависит от стиля текста и в среднем составляет 22.2 предложений/с для публицистических и художественных текстов и 66.7 предложений/с для научных текстов.

```
SyntaxParser sp = new SyntaxParser();
sp.init();
BearingPhraseSP phrase = sp.getTreeSentence("Мама мыла раму.");
System.out.println(phrase);

Output:
BearingPhraseSP{
 words=[
  WordSP=[currencyForm={hash=38116099,мама,ToS=17},
            main=null, dependents=[{hash=436508163,мыла,ToS=20}]],
  WordSP=[currencyForm={hash=436508163,мыла,ToS=20},
            main={hash=38116099,мама,ToS=17},
            dependents=[{hash=687768067,раму,ToS=17}
           ]
  ],
  WordSP=[currencyForm={hash=687768067,раму,ToS=17},
            main={hash=436508163,мыла,ToS=20}, dependents=[]
  ]
 ],
 mainOmoForm=[currencyForm={hash=38116099,мама,ToS=17},
                  main=null, dependents=[{hash=436508163,мыла,ToS=20}]
 ]
}
```

Рис. 2. Пример вызова метода семантико-синтаксического анали-за и полученного результата [Fig. 2. An example of calling the method of semantic-syntactic analysis and its result]

Вычислительная сложность алгоритма O(n), т. к. время работы алгоритма применения фильтра и алгоритма поиска связей в пределах опорного оборота близка к постоянному значению, а изменение в средней скорости работы инструмента при анализе одного предложения не зависит от объема текста.

Также, реализованный инструмент применяется для выделения смысловой неделимой части предложения — понятий. Именно понятие передает мыслительный образ (предмет, явление, событие и т. д. реального мира) [27], понятие нельзя разделить [26], например, понятие «ракетные войска стратегического назначения» отражает конкретный род войск, которые выделяются среди остальных, и если разделить это понятие на более мелкие единицы «ракетные войска» и «стратегическое назначение», то получатся два термина, которые не способны описать исходный род войск, т. е. они описывают другие объекты. А также «ракетные войска стратегического назначения» является устойчивым фразеологизмом это подтверждается тем, что для такого рода войск существует устойчивая аббревиатура — РВСН.

Понятие может иметь длину от 2 до 17 слов [19], поэтому выбираются все комбинации сочетаний слов длиной от 2 до 17 слов. Далее происходит подсчет количества встречающихся сочетаний слов, причем подсчет идет по приведенным в начальную форму омоформам, для того чтобы такие сочетания как «мыла раму» и «моем рамы» считалось как одно «мыть рама».

Происходит подсчет частоты встречаемости выделенных словосочетаний, отсекаются все сочетания, которые не могут являться словосочетанием, образуя в конечном итоге список потенциальных понятий, выделенных из текста. Затем для поиска ключевых словосочетаний происходит отсеивание словосочетаний, содержащих глаголы и не содержащих хотя бы одного имени существительного, оканчивающегося и начинающегося на предлог. Таким образом, составляется список кандидатов на понятия и словарь понятий.

4. РАЗВИТИЕ СОЗДАННЫХ АЛГОРИТМОВ И ИНСТРУМЕНТА СЕМАНТИКО-СИНТАКСИЧЕСКОГО АНАЛИЗА

Естественный язык является постоянно развивающемся, меняющимся, а русский язык к тому же имеет достаточно свободные правила к построению предложения (например, в отличие от английского или японского языка, где порядок частей речи в предложении закреплен). Это приводит к невозможности создания универсального набора правил для семантико-синтаксического анализа, тем не менее приведенный набор правил позволяет определить большинство связей. Но для поддержания актуальности реализации и для улучшения качества анализа необходимо расширять существующие правила, а также совершенствовать существующие и добавлять новые.

Одним из подвидов таких правил могут стать правила, основанные на модели управления словами для некоторых классов [31]. Модель управления присуща большинству частей речи: существительным, отглагольным существительным, глаголом и т. д., например, глагол «лежит» может управлять только существительными в им. и в т. падеже, а для слова «подшутить» модель будет такая: «над + дат. падеж». В тоже время для слова «висеть» будет уже другая модель управления: «в/на + предл. падеж» или «под/над + твор. падеж» или «у + род. падеж». Таким образом, может быть расширен набор правил, описанных в табл. 2, что позволит улучшить качество выделение словосочетаний.

Вторым из подвидов правил может выступать прагматический этап анализа, например, если два именных словосочетания находятся в отношении «род-вид», то с высокой вероятностью в таком же отношении находятся и их опорные слова [19].

Помимо разработки правил, необходимо работать над решением проблемы, которая возникает, если в предложении после фильтра снятия неоднозначности остается много неоднозначных слов, которые образуют несколько комбинаций результата анализа.

Также, важным направлением является дальнейшая оптимизация реализации алгоритмов, чтобы обеспечить возможность применения семантико-синтаксического анализа в новых областях для решения прикладных задач.

ЗАКЛЮЧЕНИЕ

Объединение синтаксического и семантического анализа в семантико-синтаксический, позволяет учитывать семантику текста при поиске синтаксических связей между словами, а также позволяет рассматривать семантический анализ без отрыва от синтаксиса. Также, выделение такого этапа анализа позволило выявить правила сочетаемости слов и разработать алгоритмы фильтра устранения неоднозначности и поиска синтаксических связей между словами.

Апробация созданного инструмента показала, что снятие неоднозначности слов доходит до 50 %, а точность установления связей между словами до 70–80 %, причем на результат влияет стиль текста. Возможным способом улучшения работы алгоритмов является добавление правил сочетаемости слов, а также решение проблемы, возникающей нескольких комбинаций результата семантико-синтаксического анализа.

Дальнейшая оптимизация алгоритмов и структур данных позволит обеспечить возможность применения семантико-синтаксического анализа в новых областях для решения прикладных задач.

КОНФЛИКТ ИНТЕРЕСОВ

Авторы декларируют отсутствие явных и потенциальных конфликтов интересов, связанных с публикацией настоящей статьи.

СПИСОК ЛИТЕРАТУРЫ

1. *Bender, E. M.* Linguistic Fundamentals for Natural Language Processing II: 100 Essentials from Semantics and Pragmatics. / E. M. Bender. – Synthesis Lectures on Human Language Technologies. – London: Morgan & Claypool, 2019 – 268 p. DOI:10.2200/S00935ED1V02Y201907HLT043

2. *Хорошилов, А. А.* Методы автоматического установления смысловой близости документов на основе их концептуального анализа / А. А. Хорошилов // Труды 15-й Всероссийской научной конференции «Электронные библиотеки: перспективные методы и технологии, электронные коллекции» – RCDL-2013, – Ярославль, 2013. – С. 20–28.

3. *Newport, E. L.* Statistical language learning: computational, maturational, and linguistic constraints / E. L. Newport // Language and Cognition. – 2016. – № 8. – P. 447–461. DOI: 10.1017/langcog.2016.20

4. Автоматическая обработка текстов на естественном языке и анализ данных: учеб. пособие / Большакова Е.И. [и др.] – М. : Изд-во НИУ ВШЭ, 2017. – 269 с

5. *Porter, M.* An algorithm for suffix stripping / M. Porter // Readings in Information Retrieval. – San Francisco, CA, 1997: Morgan Kaufmann Publishers. – 1997. – 313–316 p.

6. Официальный сайт Snowball. – Режим доступа: http://snowballstem.org. – (Дата обращения: 21.12.2019).

7. Официальный сайт Автоматическая Обработка Текста. – Режим доступа – http://aot.ru. – (Дата обращения: 21.12.2019).

8. Официальная страница Greeb. – Режим доступа – https://github.com/dustalov/greeb. – (Дата обращения: 21.12.2019).

9. Официальный сайт NLTK. – Режим доступа – http://www.nltk.org/. – (Дата обращения: 21.12.2019).

10. Официальный сайт Solarix. – Режим доступа – http://solarix.ru/. – (Дата обращения: 21.12.2019).

11. *Politsyna, E. V.* Development of the Cross-platform Library of Morphological Analysis of the Russian Language Text for Industrial Software / E. V. Politsyna, S. A. Politsyn, A. S. Porechny // CEE-SECR ‘18 Central and Eastern European Software Engineering Conference Russia Moscow, Russian Federation – October 12 – 13, 2018. – ACM New York, NY, USA, 2018. DOI: 10.1145/3290621.3290635

12. Официальная страница JMorfSdk. – Режим доступа: https://github.com/jalexpr/jmorfsdk. – (Дата обращения: 21.12.2019).

13. Официальный сайт FreeLingю. –Режим доступа – http://nlp.lsi.upc.edu/freeling/ – (Дата обращения: 21.12.2019).

14. Официальный страница MAnalyzer. – Режим доступа. – https://github.com/kmingulov/MAnalyzer. – (Дата обращения: 21.12.2019).

15. Официальный страница Russian Morphology. – Режим доступа – https://code.google.com/archive/p/russianmorphology. – (Дата обращения: 21.12.2019).

16. Официальный сайт ABBYY. Режим доступа – https://www.abbyy.com/ru-ru/isearch/compreno/. – (Дата обращения: 21.12.2019).

17. Официальный сайт GATE – General architecture for text engineering. Режим доступа – https://gate.ac.uk. – (Дата обращения: 21.12.2019).

18. *Раков, В. И.* Системный анализ (начальные понятия): учеб. пособие / Раков, В. И. – М. : Изд-во Академия Естествознания, 2012. – 239 с.

19. *Белоногов, Г. Г.* Теоретические проблемы информатики. Том 2. Семантические проблемы информатики / Под общей редакцией К. И. Курбакова. – М. : КОС•ИНФ, РЭА им Г. В. Плеханова, 2008. – 223 с.

20. *Gildea, D.* Ordered Tree Decomposition for HRG Rule Extraction / D. Gildea, G. Satta, X. Peng // Computational Linguistics. – 2019. – V. 45, No 2. – P. 339–379. DOI: 10.1162/COLI_a_00350

21. *Bach, E.* The Case for Case. / E. Bach and Harms R. T. – Universals in Linguistic Theory. 1968. – 88 p.

22. *Попов, Э. В.* Общение с ЭВМ на естественном языке / Э. В. Попов. – М. : Наука. Главная редакция физико-математической литературы, 1982. – 360 с

23. *Евдокимова, Е. С.* Естественно-языковые системы. Курс лекций / Е. С. Евдокимова. – Улан-Удэ.: ВСГТУ, 2006. – 92 с.

24. *Бэкон, Ф.* Новый Органон; [пер. англ. С. Красильщикова; вступит. ст. Б. Подороги]. – М. : РИПОЛ классик, 2019. – 364 с.

25. *Tripodi, R.* A Game-Theoretic Approach to Word Sense Disambiguation. / R. Tripodi, M. Pelillo // Computational Linguistics. – 2014. – V. 43, No 1. – P. 31–70. DOI: 10.1162/COLI_a_00274.

26. *Tsvetkov, Y.* Identification of Multiword Expressions by Combining Multiple Linguistic Information Sources / Y. Tsvetkov, S. Wintner // Computational Linguistics. – 2014. – V. 40, No 2. – P. 449–468. DOI: 10.1162/COLI_a_00177

27. *Белоногов, Г. Г.* Автоматизация составления и ведения словарей для систем фразеологического машинного перевода текстов с русского языка на английский и с английского на русский / Г. Г. Белоногов [и др.] // Научно-техническая информация, Серия 2. Выпуск № 12. – М. : ВИНИТИ, 1993. – С. 16–21.

28. *Белоногов, Г. Г.* Интерактивная система русско-английского и англо-русского машинного перевода политематических научно-технических текстов / Г. Г Белоногов. [и др.] // Научно-техническая информация, Серия 2. Выпуск № 12. – М. : ВИНИТИ, 1993. – С. 20–27.

29. *Politsyna, E. V.* The Framework for Hypothesis Verification and Analysis of Natural Language Processing for the Russian Language / E. V. Politsyna, S. A. Politsyn, A. S. Porechny // Supplementary Proceedings of the Seventh International Conference on Analysis of Images, Social Networks and Texts (AIST-SUP 2018), Moscow, Russia, July 5–7, 2018. – CEUR Work-Shop Proceedings, Aachen, Germany, 2018. – V. 2268. – P. 25–33.

30. Официальная страница фреймворка TAWT. – Режим доступа: https://github.com/jalexpr/TAWT. – (Дата обращения: 21.12.2019).

31. *Hellan, L.* Contrastive Studies in Verbal Valency / L. Hellan, A. Malchukov, M. Cennamo. – Linguistik Aktuell/Linguistics Today. – Amsterdam: John Benjamins Publishing Company, 2017 – 484 p. DOI: 10.1075/la.237

Поречный Александр Сергеевич – аспирант кафедры 319 Московского авиационного института (национального исследовательского университета).
E-mail.ru: alex.porechny@mail.ru
ORCID iD: https://orcid.org/0000-0003-2280-7406

DOI: https://doi.org/10.17308/sait.2020.1/2630 ISSN 1995-5499
Received 23.12.2019
Accepted 15.03.2020

DEVELOPMENT OF A TOOL FOR THE SEMANTIC AND SYNTACTIC ANALYSIS OF TEXTS IN RUSSIAN

© 2020 A. S. Porechny✉

Moscow Aviation Institute (National Research University)
4, Volokolamskoe sh., 125993 Moscow, Russian Federation

Abstract. Natural language processing can be divided into several levels. However, if considered separately, these levels are difficult to analyse, and the difficulties occurring on one level can only be resolved on the next one. This is especially apparent on the syntactic level, since the connections between words are determined by the "meaning" of the text, i.e. the level of semantics. Therefore we propose to combine the stages of syntactic and semantic analysis into a single stage. Semantic and syntactic analysis allows considering the semantics at the level of syntactic analysis, thereby reducing ambiguity of the text and improving the quality of the analysis. In order to implement the analysis algorithm, we determined the rules for establishing relationships between words, and developed algorithms for eliminating the ambiguity of words and search for relationships within key phrases and sentences. The article also presents the results of the implementation of the suggested semantic and syntactic analysis algorithm.
Keywords: natural language processing (NLP), parsing, semantic analysis, semantic and syntactic analysis, disambiguation filter, key words, key phrases.

CONFLICT OF INTEREST

The authors declare the absence of obvi-ous and potential conflicts of interest related to the publication of this article.

REFERENCE

1. *Bender E. M.* Linguistic Fundamentals for Nat-ural Language Processing II: 100 Essen-tials from Semantics and Pragmatics. Synthesis Lectures on Human Lan-guage Tech-nologies. London: Morgan & Clay-pool, 2019. 268 p. DOI:10.2200/S00935ED1V02Y201907HLT043.

2. *Khoroshilov A. A.* Methods for auto-matically establishing the semantic proximity of documents based on their conceptual analysis. Trudy 15-j Vserossijskoj nauchnoj konferencii «Elektronnye biblioteki: perspektivnye metody i tekhnologii, elektronnye kollekcii». Russian, Ya-roslavl, 2013. pp. C. 20–28. (in Russian)

3. *Newport E. L.* Statistical language learning: computational, maturational, and lin-guistic constraints. Language and Cogni-tion. 2016. No 8. Pp. 447–461. DOI: 10.1017/langcog.2016.20

4. *Bolshakova E. I., Vorontsov K. V., Efremova N. E., Klyshinsky E. S., Lukashevich N. V., Sapin A. S.* Automatic processing of texts in a natural language and data analysis: textbook. Russia, Moscow, 2017, 269 p. (in Russian)

5. Porter M. An algorithm for suffix stripping. Readings in Information Retrieval. San Francisco, CA, 1997: Morgan Kaufmann Publishers. 1997. 313–316 p.

6. Official site of Snowball. Available at http://snowballstem.org (accessed 21.12.2019).

7. Official site of Automatic Text Pro-cessing. Available at http://aot.ru (accessed 21.12.2019).

8. Official site of Greeb. Available at https://github.com/dustalov/greeb (accessed 21.12.2019).

9. Official site of NLTK. Available at http://www.nltk.org (accessed 21.12.2019).

10. Official site of Solarix. Available at http://solarix.ru/ (accessed 21.12.2019).

11. *Politsyna E. V.* Development of the Cross-platform Library of Morphological Analysis of the Russian Language Text for Industrial Software. CEE-SECR '18 Central and Eastern

✉ Porechny Alexandr S.
e-mail: alex.porechny@mail.ru

European Software Engineering Conference Russia Moscow, Russian Federation, October 12–13, 2018. ACM New York, NY, USA, 2018. DOI: 10.1145/3290621.3290635

12. Official site of JMorfSdk. Available at https://github.com/jalexpr/jmorfsdk (accessed 21.12.2019).

13. Official site of FreeLingю. Available at http://nlp.lsi.upc.edu/freeling (accessed 21.12.2019).

14. Official site of MAnalyzer. Available at https://github.com/kmingulov/MAnalyzer (accessed 21.12.2019).

15. Official site of RussianMorphology. Available at https://code.google.com/archive/p/russianmorphology (accessed 21.12.2019).

16. Official site of ABBYY. Available at https://www.abbyy.com/ru-ru/isearch/compreno (accessed 21.12.2019).

17. Official site of GATE – General archi-tecture for text engineering. Available at https://gate.ac.uk (accessed 21.12.2019).

18. *Rakov V. I.* System analysis (initial concepts): textbook. allowance. Russia, Mos-cow, 2012, 239 p. (in Russian)

19. *Belonogov G. G.* Theoretical problems of computer science. Volume 2. Semantic prob-lems of computer science. KOS INF Plekhanov Russian University of Economics, 2008, 223 p. (in Russian).

20. *Gildea D.* Ordered Tree Decomposi-tion for HRG Rule Extraction. Computational Linguistics. 2019. V. 45, No 2. Pp. 339–379. DOI: 10.1162/COLI_a_00350

21. *Bach E., Harms R. T.* The Case for Case. Universals in Linguistic Theory. 1968. 88 p.

22. *Popov E. V.* Communication with computers in a natural language. Russia, Mos-cow, 1982, 360 p. (in Russian)

23. *Evdokimova, E. S.* Natural language systems. Lecture course. Russia, Ulan-Ude, 2006. 92 p. (in Russian)

24. *Bacon F.* New Organon; [trans. English S. Krasilshchikova; will enter. Art. B. Sub-limates]. Russia, Moscow, 2019. 364 p.

25. *Tripodi R., Pelillo M.* A Game-Theoretic Ap-proach to Word Sense Disambiguation. Computational Linguistics. 2014. V. 43, No 1. Pp. 31–70. DOI: 10.1162/COLI_a_00274

26. *Tsvetkov Y., Wintner S.* Identification of Multi-word Expressions by Combining Multiple Lin-guistic Information Sources. Computational Linguistics. 2014. V. 40, No 2. Pp. 449–468. DOI: 10.1162/COLI_a_00177

27. *Belonogov G. G., Zelenkov Y. G., Kuznetsov B. A., Novoselov A. P., Khoroshilov A. A., Khoroshilov A. A.* Automation of collecting and maintaining dictionaries for sys-tems of phraseological computer translation of texts from Russian into English and from Eng-lish to Russian. Nauchno-tehnicheskaja infor-macija, 1993. vol. 2. No.12. pp. 16-21. (in Russian).

28. *Belonogov G. G., Zelenkov Ju. G., Kuznecov B. A., Novoselov A. P., Pashhenko N. A., Horoshilov A. A., Horoshilov A. A.* An interactive system of Russian-English and Eng-lish-Russian machine translation of polythematic scientific and technical texts. Nauchno-tehnicheskaja informacija, 1993. V. 2, No.3. Pp. 20–27. (in Russian).

29. *Politsyna E. V., Politsyn S. A., Porechny A. S.* The Framework for Hypothesis Verification and Analysis of Natural Language Processing for the Russian Language. Supplementary Proceedings of the Seventh International Conference on Analysis of Images, Social Networks and Texts (AIST-SUP 2018), Moscow, Russia, July 5–7, 2018. CEUR Work-Shop Proceedings, Aachen, Germany, 2018. v. 2268. Pp. 25–33.

30. Official site of framework TAWT. Available at https://github.com/jalexpr/TAWT (accessed 21.12.2019).

31. *Hellan L., Malchukov A., Cennamo M.* Contrastive Studies in Ver-bal Valency. Linguistik Aktuell/Linguistics Today. Amsterdam: John Benjamins Publishing Com-pany, 2017. 484 p. DOI: 10.1075/la.237

Porechny Alexandr S. — postgraduate student, Department No. 319, Moscow Aviation Institute (National Research University).
E-mail: alex.porechny@mail.ru
ORCID iD: https://orcid.org/0000-0003-2280-7406