

УДК 681.3.07

А.А. Иванков, А.Л. Шишагин

**РАСШИРЕНИЕ ОБЪЕКТНОЙ МОДЕЛИ СТЭНФОРДСКОГО ПАРСЕРА
ДЛЯ РЕШЕНИЯ ЗАДАЧИ ИДЕНТИФИКАЦИИ
СЕМАНТИЧЕСКИХ ТРИПЛЕТОВ**

А.А. Ivankov, A.L. Shishagin

STANFORD NLP PARSERS AND SEMANTIC TRIPLETS IDENTIFICATION

Изучен информационный поиск семантических триплетов в текстах англоязычного корпуса Википедии. Для идентификации триплетов использован стэнфордский парсер, объектная модель которого расширена авторами. Приведены оценки вероятности ошибки идентификации триплета.

ИНФОРМАЦИОННЫЙ ПОИСК. СЕМАНТИЧЕСКИЙ АНАЛИЗ. СТЭНФОРДСКИЙ ПАРСЕР. СЕМАНТИЧЕСКИЙ ТРИПЛЕТ. ОШИБКИ ИДЕНТИФИКАЦИИ.

We developed some own classes for the object model of Stanford NLP parsers to implement our algorithm of semantic triplet identification. Results of information retrieving are discussed for en.Wikipedia articles.

INFORMATION RETRIEVING. SEMANTIC ANALYSIS. STANFORD NLP PARSERS. SEMANTIC TRIPLET. RETRIEVING ERRORS.

В работе [1], посвященной семантическому анализу и оцениванию различных характеристик документов на основе результатов такого анализа, алгоритмы ориентированы преимущественно на использование элементов метаданных, т. е. элементов разметки, определяющих структуру документа. Мы анализировали текст на уровне заголовков, подзаголовков и т. п., используя соответствующую грамматику языка разметки конкретного документа (в нашей работе это были подмножества грамматики HTML/XML) и модель «мешка слов».

Между тем структурообразующие элементы содержат существенно меньшую часть информации, чем предложения дискурсов в тексте самого документа. Задача семантического анализа не может быть решена в полном объеме, если не выполня-

ется семантический анализ каждого предложения дискурса.

Большинство лингвистов разделяют следующую точку зрения: актуальное членение предложения на естественном языке (ЕЯ) предполагает выделение особого уровня (отличного от синтаксического), на котором грамматическими средствами выражается субъектно-предикатная структура соответствующей мысли. На этом, особом, логико-грамматическом уровне предложения мы имеем дело с логико-грамматическим субъектом и логико-грамматическим предикатом, которые не обязательно должны совпадать с синтаксическими подлежащим и сказуемым. Но в то же время давно замечено [2], что в научно-технической литературе такое соответствие имеет место почти всегда, если привлекать язык статистики.

В пределах отдельного предложения наибольший вклад в семантику этой элементарной части дискурса вносится триплетом (тройкой): подлежащее — сказуемое — дополнение. Научно-технические тексты характеризуют преимущественно как номинативные, но особую роль в них играет сказуемое. Глагол или существительное как часть составного сказуемого — это смыслообразующие элементы фразы (главные ее элементы) [3–5]. Более того, глагол-сказуемое поддерживает тему-рематическую прогрессию текста, что принципиально важно в рамках коммуникативно-функционального подхода к построению модели языка. Напомним также, что язык RDF (Resource Definition Framework) создан консорциумом W3 как средство семантического описания интернет-ресурсов, ориентированное на триплеты: субъект — предикат — объект.

Следовательно, одной из важнейших задач семантического анализа предложений является идентификация семантических триплетов.

Хотелось бы еще заметить, что эту задачу ни в коем случае не следует рассматривать как идентификацию триграмм в вероятностных порождающих моделях документов. В последнем случае для решения, вообще говоря, не требуется ни синтаксический анализ предложения с построением дерева разбора, ни последующий семантический анализ с идентификацией грамматических отношений.

Попытки решить задачу идентификации семантических триплетов в пределах предложения документа произвольного корпуса предпринимались неоднократно. По мере развития методов информационного поиска эта проблема вызывает все больший интерес. Информация о структуре и местоположении триплета в пределах предложения может использоваться для решения самых актуальных проблем обработки текстов на ЕЯ: их автоматического аннотирования, машинного перевода и т. п. Например, описание корпоративных хранилищ информации предлагают в форме семантических графов (на основе дерева синтаксического разбора, снабженного разметкой грамматических отношений лексем) как одного из

представлений семантической сети. Такое представление является весьма наглядным. Другое место приложения результатов решения этой задачи — автоматическое построение онтологий предметных областей и, разумеется, семантическое описание интернет-ресурсов с использованием языка RDF.

Основные трудности в решении связаны с неоднозначностью результатов синтаксического и семантического анализа предложения. Операторы синтаксического разбора порождают нам дерево (граф) структуры предложения. Дальнейший анализ выполняется с помощью оператора семантического анализа, который снабжает дерево синтаксического разбора разметкой грамматических отношений лексем, входящих в него.

Результаты, доставляемые нам этими операторами в конечном итоге, — это лес случайных деревьев. Математические модели на классе случайных помеченных деревьев на практике приводят к ресурсоемким алгоритмам оценивания. Наш опыт работы с подобными моделями позволяет нам утверждать, что получение отдельных оценок на корпусе неаннотированных документов может потребовать до нескольких недель работы вычислительной системы.

Тем самым становится очевидным, что выбор инструментария для реализации алгоритмов решения задачи — ключевой момент в таких исследованиях.

Множество основных критериев выбора содержит по меньшей мере следующие:

- форматы хранения входных данных и результатов;
- портируемость решения на другие платформы;
- язык реализации.

Большинство синтаксических анализаторов для представления результатов синтаксического анализа предложений используют формат Penn TreeBank [6] (рис. 1). Такая форма представления дерева синтаксического разбора стала стандартом де-факто для парсеров, работающих с текстами на английском языке. Это весьма удивительный факт, принимая во внимание, что в отношении других форматов эти

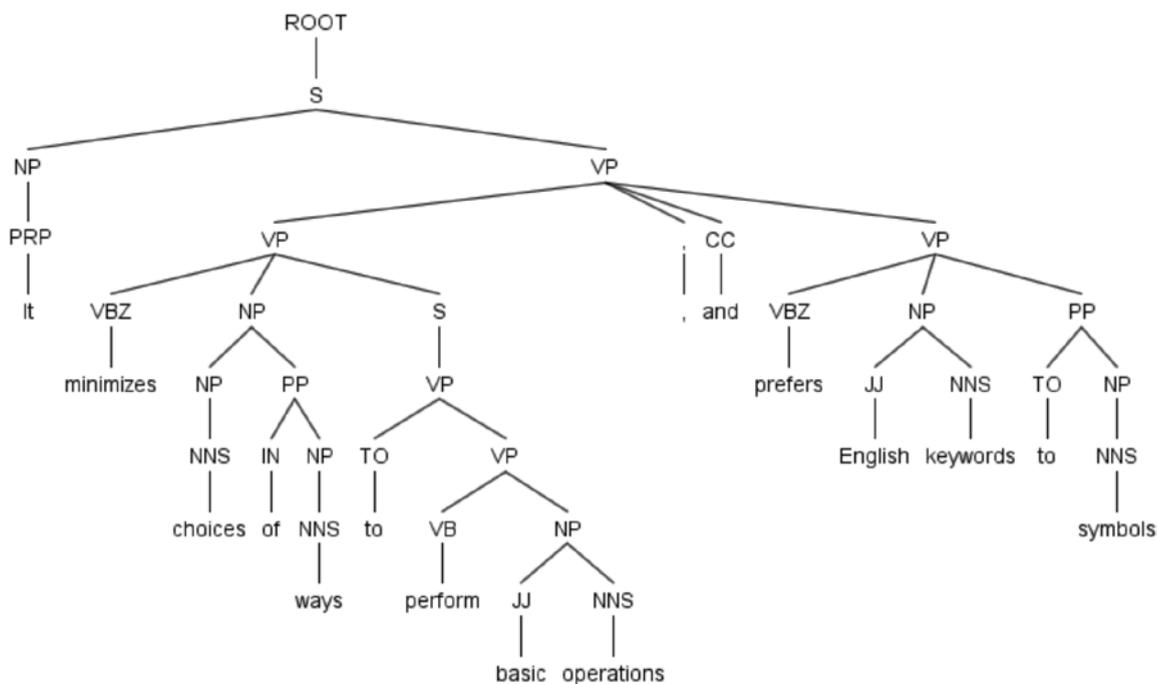


Рис. 1. Графическое представление дерева синтаксического разбора

программные системы отличаются весьма значительно. Отметим, что существует некоторое небольшое подмножество парсеров, где выходной формат ориентирован на представление грамматических зависимостей.

Основное системное требование – программное обеспечение должно легко разворачиваться на платформе WINTEL или в одной из популярных Unix-систем. С точки зрения портируемости программного обеспечения исследователи отдают предпочтение решениям на объектно-ориентированном языке программирования JAVA. Наиболее доступные решения с открытым JAVA-кодом уже более 10 лет предлагаются на сайтах <http://opennlp.apache.org> и <http://sourceforge.net/projects/opennlp>.

Принимая во внимание указанные выше критерии, наш выбор осуществлялся из небольшого множества анализаторов с открытым кодом: LinkParser, MSTParser, MultiParser, Stanford Core NLP.

Необходимость идентификации грамматических отношений в пределах отдельного дерева синтаксического разбора – это

еще одно функциональное требование к инструментарию. Номенклатура грамматических отношений (отношений зависимости, определяемых на узлах дерева синтаксического разбора) существенно отличается от парсера к парсеру. Наименьшая мощность множества грамматических отношений у консервативной схемы [7]. Парсер LinkParser напротив предлагает исследователю более тонкую структуру этого множества. Мощность множества и качество идентификации ее элементов в тексте на ЕЯ – еще два критерия выбора инструмента. Они не взаимодополняющие: выбор в пользу одного может привести к потерям в отношении другого.

В литературе отмечается, что более высокое качество идентификации грамматических отношений демонстрируют парсеры, реализующие алгоритм поиска соответствий между отдельными подграфами дерева синтаксического разбора и абстрактными шаблонами, хранимыми как структуры данных в словарях этой программной системы.

Помимо критериев портируемости и

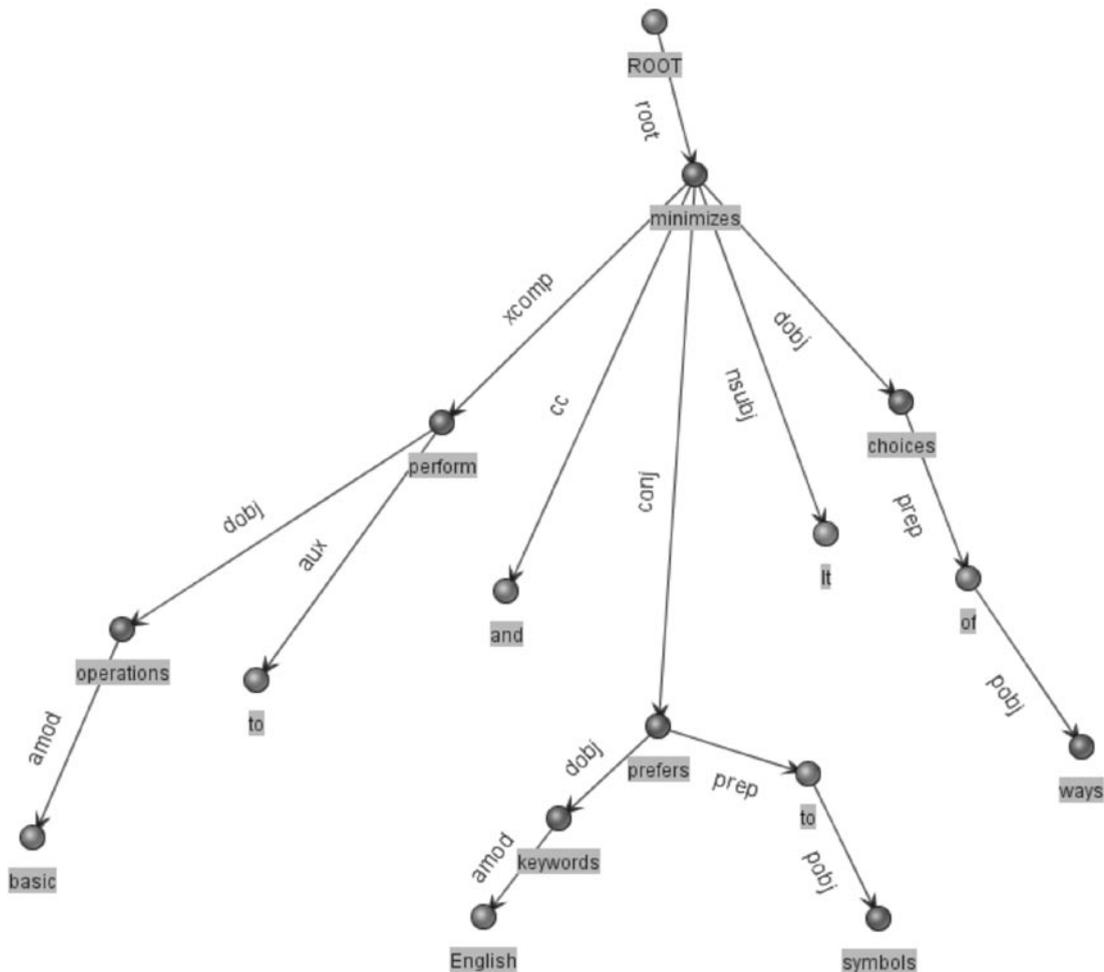


Рис. 2. Графическое представление результатов семантического анализа отдельного предложения

языка реализации, предпочтение отдавали хорошо документированному исходному коду.

Мы остановили свой выбор на пакете Stanford Core NLP [8] по следующим соображениям:

- структуры дерева синтаксического разбора совпадают с классическим форматом Penn TreeBank;
- множество идентифицируемых грамматических отношений расширено по сравнению с классической схемой [7], добавлены дополнительные отношения (см. рис. 2), играющие важную роль, обеспечивая успех последующего семантического анализа;
- форматы этого анализатора уже реализованы разработчиками других семантических анализаторов, т. е. апробированы и

хорошо зарекомендовали себя на практике;

- семантический анализ можно параметризовать с помощью шаблонов, определяющих правила разбора;
- исходный код доступен под лицензией GPL;
- кроссплатформенный язык разработки – JAVA, на этот язык уже портировали и ряд других анализаторов.

Наибольшие сложности породило отсутствие качественной документации. Единственно эффективным инструментом для изучения пакета как всегда оказался отладчик. Библиотеку из 467 классов, около 150 000 строк исходного кода, мы подвергли рефакторингу, исключив «грязную» сборку, и дополнительно разработали 45 классов, реализующих недостающую в

исходном пакете функциональность.

Все другие модификации кода были вынужденной мерой в условиях ограниченности аппаратных ресурсов, которыми мы располагали для проведения исследования. Заметим, что одни лишь словари парсера требовали более 7Г адресного пространства, не говоря уже об объеме собственно байт-кода.

Идентификация триплетов

В ходе анализа публикаций, посвященных семантическому анализу текстов на английском языке, наше внимание привлекли работы, упоминаемые в [9]. Это, пожалуй, один из немногих апробированных подходов к решению задачи идентификации триплетов, алгоритм которого опубликован.

Решение задачи авторы излагают в виде списка правил, применяемых к дереву синтаксического разбора предложения. Структура этого дерева (S в нотации Penn TreeBank) содержит два дочерних элемента: NP – именное словосочетание; VP – глагольное словосочетание. Поиск подлежащего (SUBJ) выполняется в пределах NP – первый дочерний элемент NP, грамматическая категория «часть речи» которого указана как «существительное», атрибутируется как подлежащее (SUBJ). Поиск сказуемого (PRED) осуществляется в пределах VP – наиболее удаленный по уровню иерархии терминальный элемент, грамматическая категория «часть речи» которого указана как «глагол», атрибутируется как сказуемое (PRED).

Наконец, поиск дополнения осуществляется в элементах того же уровня иерархии (по отношению к S), что и VP. По меньшей мере три различных поддерева рассматриваются как потенциальные контейнеры дополнения ((OBJ)):

словосочетание, содержащее предлог (PP);

именное словосочетание (NP);

словосочетание с прилагательным (ADJP).

В первых двух типах словосочетаний осуществляют поиск первого вхождения существительного, в последнем случае – вхождение первого прилагательного.

К сожалению, авторы не привели количественных оценок качества идентификации, ограничившись иллюстрацией работы парсеров на десяти английских предложениях.

Мы реализовали алгоритм идентификации, предложенный в [9] и последующих работах автора, в частности, опубликованных в журнале *Informatica*. Разумеется, наш программный код может отличаться от авторской версии, а в личной переписке с авторами мы не смогли получить от них необходимых нам пояснений, позволяющих утверждать, что наша реализация идентична их реализации.

Тестирование выполнялось на статьях англоязычного корпуса *Wikipedia*, срез на 01.01.2011. Наш выбор обусловлен тем, что в большинстве работ, где обсуждается качество идентификации грамматических отношений и синтаксического анализа, оценки получают на текстах, составленных профессионалами, журналистами СМИ. Мы же хотим получить оценки на текстах «живого» языка, на текстах, которые практически не подвергались редакторской правке.

Результаты тестирования указывают на то, что предложенные авторами [9] приводят к потере большей части семантических триплетов. Рассмотрим итоги разбора первых предложений статьи «Ada (programming language)».

Применение суперпозиции операторов синтаксического и семантического разбора порождает результат, который выглядит следующим образом (скобочная запись):

(ROOT (S (NP (NNP Ada)) (VP (VBZ is) (NP (NP (DT a) (ADJP (JJ structured) (,) (JJ statically)) (JJ typed) (,) (ADJP (JJ imperative) (,) (JJ wide-spectrum) (,) (CC and) (JJ object-oriented) (JJ high-level)) (NN computer) (VBG programming) (NN language)) (,) (VP (VBN extended) (PP (IN from) (NP (NP (NNP Pascal)) (CC and) (NP (JJ other) (NNS languages)))))) (.)

Подлежащее идентифицировано успешно: (NNP Ada). В нотации [9] – subj, в оригинальной стэнфордской нотации – nsubj. Однако сказуемое (NN language) было утеряно, а вместо него в качестве предиката предложено причастие (VBN extended).

В качестве дополнения – (NNP Pascal) – ошибочная интерпретация предложного дополнения из причастного оборота.

Еще один пример:

(ROOT (S (NP (NNP Ada)) (VP (VBD was) (ADVP (RB originally)) (VP (VBN designed) (PP (IN by) (NP (NP (DT a) (NN team)) (VP (VBN led) (PP (IN by) (NP (NP (NNP Jean) (NNP Ichbiah)) (PP (IN of) (NP (NP (NNP CII) (NNP Honeywell) (NNP Bull)) (PP (IN under) (NP (NN contract)))))))) (PP (TO to) (NP (NP (DT the) (NNP United) (NNPS States) (NNP Department)) (PP (IN of) (NP (NP (NNP Defense)) (PRN (-LRB- -LRB-) (NP (NNP DoD)) (-RRB- -RRB-)))))) (PP (IN from) (NP (CD 1977) (TO to) (CD 1983))) (S (VP (TO to) (VP (VB supersede) (NP (NP (DT the) (NNS hundreds)) (PP (IN of) (NP (NP (NN programming) (NNS languages)) (VP (ADVP (RB then)) (VBN used) (PP (IN by) (NP (DT the) (NNP DoD)))))))))))))) (.)

Подлежащее и в этом предложении было успешно идентифицировано: (NNP Ada). Сказуемое (VBN designed) утеряно, а в его качестве предложено (VBN used) из придаточного предложения.

Мы убедились, что правила идентификации семантических триплетов из [9] приводят к потере значимой информации.

Приняв во внимание результаты тестирования правил идентификации триплетов, изложенных в [9], был предложен собственный алгоритм. Он (алгоритм идентификации триплетов) определяется набором структурных и грамматических правил, которые на уровне реализации хранятся во внешнем файле-шаблоне. Эти правила последовательно применяются ко всему дереву синтаксического разбора или отдельным его частям.

Правила идентификации семантических триплетов

Алгоритм реализован на основе набора правил, которые применяются к дереву, полученному по итогам разбора структуры предложения, и к семантическому дереву отношений:

SR (Structure Rules) – структурные правила;

GR (Grammar Rules) – грамматические правила.

Структурные правила служат для идентификации в пределах одного предложения его придаточных предложений, причастных и деепричастных оборотов, отдельных фраз. Таким образом мы определим, в каких частях предложения содержатся объекты «семантический триплет» (далее – ST) из одной группы.

К структурным правилам мы отнесли следующие:

1. Одной семантической группе принадлежат все терминальные узлы дерева структурного разбора и соответствующие им объекты ST, которые имеют одинаковый ближайший вверх по дереву узел S (Sentence) или (Root).

2. Одной семантической группе принадлежат все терминальные узлы дерева структурного разбора и соответствующие им объекты ST, которые имеют одинаковый ближайший вверх по дереву узел VP или NP. В этом случае на узлы VP и NP накладываются дополнительные ограничения: их родительский узел должен быть S и среди дочерних узлов S исходные узлы должны следовать после первого вхождения узла VP.

3. Если в поддереве с корневым узлом PP обнаружено поддерево PP, то в нем игнорируются все поддерева, кроме тех, у которых корневой узел S.

Основное грамматическое правило определяет обобщения и фильтрации грамматических отношений:

1. Из всех типов грамматических зависимостей, которые имеются в иерархическом дереве стэнфордского парсера, для построения наших правил оставлены и обобщены следующие грамматические отношения:

PRED = {pred, aux, auxpass, cop }

SUBJ = { subj, nsubj, nsubjpass, csubj, csubjpass, xsubj }

OBJ = { conj, prep, partmod, obk, iobj, dobj, pobj }

Помимо этого, определяются несколько правил включения очередного объекта ST на основе уже определенных в группе объектов. Эти аспекты поиска триплетов будут изложены в другой нашей работе.

Результаты такого обобщения положительные. В первом из указанных выше предложений подлежащее – (NNP Ada), сказуемое – (NN language). Во втором примере подлежащее – (NNP Ada), обобщенное сказуемое – (VBN designed) совместно с (VBD was), предложное дополнение – (NN team). Но не все результаты так же хорошо согласуются с экспертной оценкой. Приведем еще один характерный пример (в нашей XML нотации):

```

<sentence>
  <text>It has strong built-in language
  support for explicit concurrency, offering tasks,
  synchronous message passing, protected objects
  and nondeterminism. </text>
  <tokens>
    <token> <value > It</value>
  <role>nsubj</role> </token>
    <token> <value > has</value>
  <role>pred</role> </token>
    <token> <value > strong</value> </
  token>
    <token> <value > built-in</value>
  </token>
    <token> <value > language</value>
  </token>

```

```

    <token> <value > support</value>
  <role>dobj</role> </token>
    <token> <value > for</value>
  <role>prep</role> </token>
    <token> <value > explicit</value>
  </token>
    <token> <value > concurrency</
  value> <role>pobj</role> </token>
    <token> <value > ,</value> </to-
  ken>
    <token> <value > offering</value>
  </token>
    <token> <value > tasks</value>
    <role err=Ошибочная иденти-
  фикация грамматического отношения >conj</
  role>
    </token>
    ...
    <token> <value > synchronous</
  value> </token>
    <token> <value > message</value>
  </token>
    <token> <value > passing</value>
    <role err=Ошибочная иденти-
  фикация грамматического отношения >conj</
  role>
    </token>

```

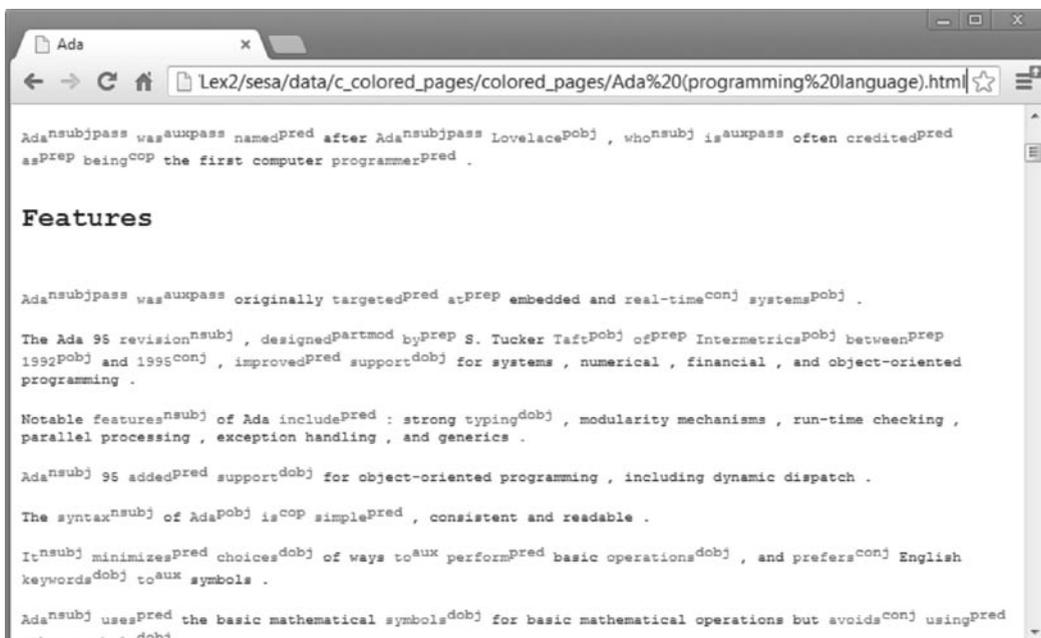


Рис. 3. Форма представления результатов синтаксического и семантического анализа статьи для выполнения ручной разметки экспертом

Оценивание уровня ошибок в результатах

Обобщенное грамматическое отношение	Общее количество	Количество ошибочно идентифицированных	Процент ошибок, %
OBJ	3214	607	18,89
PRED	1835	222	12,1
SUBJ	857	114	13,3
Сумма	5906	943	15,97

```

...
<token> <value > protected</value>
</token>
<token> <value > objects</value>
<role err=Ошибочная идентификация грамматического отношения > conj</role>
</token>
<token> <value > and</value> </token>
<token> <value > nondeterminism</value>
<role err=Ошибочная идентификация грамматического отношения > conj</role>
</token>
...
</tokens>
</sentence>

```

Обеспечив результаты синтаксического и семантического анализа HTML разметкой, мы предоставили эти материалы экспертам в виде (рис. 3), пригодном для выполнения ручной разметки этих же статей.

Ошибки в идентификации грамматических отношений лексемы очень критичны для последующего семантического анализа. Следует заметить, что в данном примере мы выбрали из случайного леса деревьев, порождаемых стэнфордским парсером, дерево, оптимальное по критерию согласия, построенному авторами пакета на основе модели порождающих грамматик, они в бинарной форме хранятся в этой программной системе (файл объемом более 7Г, загружаемый в адресное пространство приложения в ходе его работы).

В нашей работе предпринята попытка провести семантический анализ текстов англоязычной Википедии, т. е. текстов, которые подготовлены различными специалистами, не являющимися профессиональными журналистами или писателями. Мы подчеркиваем это обстоятельство, т. к. в большинстве других работ (в т. ч. в [9]) использовались размеченные экспертами журнальные статьи. Расхождения результатов идентификации семантических триплетов, полученных с помощью нашего алгоритма и алгоритма из [9] в нашей интерпретации, позволяют предположить, что в [9] и последующих работах этой группы исследователей изложен подход, нуждающийся в существенном дополнении как с точки зрения статистической обработки текстов на ЕЯ, так и с точки зрения лингвистики.

Другую причину высокого процента ошибок идентификации триплетов и, как следствие, определения семантики текста, мы видим в том, что в стэнфордском парсере грамматические отношения определяются с помощью порождающих грамматик, но с минимальным привлечением статистических методов. Первоочередные задачи в данном контексте могут быть сформулированы как выработка решающих правил (в статистическом смысле) идентификации грамматических отношений лексем, которые позволили бы минимизировать ошибку в результатах идентификации триплетов. Возможно, в поисках смысла текстов на ЕЯ более обнадеживающими будут результаты привлечения модели случайного леса деревьев. Это направление исследований представляется нам одним из наиболее перспективных.

СПИСОК ЛИТЕРАТУРЫ

1. **Иванков, А.А.** Динамическое отслеживание модулем информационно-поисковой системы изменений в структуре или тексте интернет-

ресурса [Текст] / А.А. Иванков, Д.С. Елисеев // Научно-технические ведомости СПбГПУ. Информатика. Телекоммуникации. Управление

ние. –СПб.: Изд-во Политехн. ун-та, 2010. –№ 3 (101). –С. 86–93.

2. **Пумпянский, А.Л.** Информационная роль порядка слов в научной и технической литературе [Текст] / А.Л. Пумпянский. –М.: Наука, 1974. – 247 с.

3. **Апресян, Ю.Д.** Лексическая семантика. Синонимические средства языка [Текст] / Ю.Д. Апресян. – М.: Наука, 1974. – 367 с.

4. **Филлмор, Ч.** Основные проблемы лексической семантики [Текст] / Ч. Филлмор // Новое в зарубежной лингвистике. – М.: Радуга, 1983. –Вып. XII. Прикладная лингвистика. – С. 74–122.

5. **Сильницкий, Г.Г.** Соотношение глагольных признаков различных уровней в английской

речи [Текст] / Г.Г. Сильницкий, С.Е. Андреев, Л.А. Кузьмин, М.И. Кусков. – Минск: Наука и техника, 1990. – 182 с.

6. [Электронный ресурс] / Режим доступа: <http://www ldc.upenn.edu/Catalog/docs/LDC99T42/prsguid1.pdf>

7. Treebanks: Building and Using Parsed Corpora [Text] / Ed. A. Abeille. – Springer, 2003. – P. 302–312.

8. [Электронный ресурс] / Режим доступа: <http://nlp.stanford.edu>

9. **Rusu, Delia.** Semantic Graphs Derived from Triplets with Application in Document Summarization [Text] / Delia Rusu [et al.] // Informatica. – 2009. –Vol. 33. P. 357–362.

REFERENCES

1. **Ivankov A.A., Eliseev D.S.** Dinamicheskoe otslezhivanie modulem informatsionno-poiskovoi sistemy izmenenii v strukture ili tekste Internet-resursa / Nauchno-tehnicheskie vedomosti SPbGPU. Informatika. Telekommunikatsii. Upravlenie. –St-Petersburg: Izd-vo Politehn. un-ta, 2010. –№ 3 (101). –S. 86–93. (rus)

2. **Pumpianskii A.L.** Informatsionnaia rol' poriadka slov v nauchnoi i tekhnicheskoi literature. – Moscow: Nauka, 1974. – 247 s. (rus)

3. **Apresian Iu.D.** Leksicheskaiia semantika. Sinonimicheskie sredstva iazyka. – Moscow: Nauka, 1974. – 367 s. (rus)

4. **Fillmor Ch.** Osnovnye problemy leksicheskoi semantiki / Novoe v zarubezhnoi lingvistike.

– Moscow: Raduga, 1983. –Vyp. XII. Prikladnaia lingvistika. – S. 74–122. (rus)

5. **Sil'nitskii G.G., Andreev S.E., Kuz'min L.A., Kuskov M.I.** Sootnoshenie glagol'nykh priznakov razlichnykh urovnei v angliiskoi rechi. – Minsk: Nauka i tekhnika, 1990. – 182 s.

6. <http://www ldc.upenn.edu/Catalog/docs/LDC99T42/prsguid1.pdf>

7. Treebanks: Building and Using Parsed Corpora; Ed. A. Abeille. – Springer, 2003. – P. 302–312.

8. <http://nlp.stanford.edu>

9. **Delia Rusu et al.** Semantic Graphs Derived from Triplets with Application in Document Summarization / Informatica. –2009. –Vol. 33. –P. 357–362.

ИВАНКОВ Алексей Александрович – доцент кафедры прикладной математики Санкт-Петербургского государственного политехнического университета, кандидат физико-математических наук.

195251, Россия, Санкт-Петербург, ул. Политехническая, д. 29.

E-mail: ivankov@stu.neva.ru

IVANKOV, Alexei A. St. Petersburg State Polytechnical University.

195251, Politekhnikeskaya Str. 29, St.-Petersburg, Russia.

E-mail: ivankov@stu.neva.ru

ШИШАГИН Алексей Леонидович – выпускник кафедры прикладной математики Санкт-Петербургского государственного политехнического университета, магистр.

195251, Россия, Санкт-Петербург, ул. Политехническая, д. 29.

SHISHAGIN, Alexei L. St. Petersburg State Polytechnical University.

195251, Politekhnikeskaya Str. 29, St.-Petersburg, Russia.