

МЕТОДЫ ИСПОЛЬЗОВАНИЕ ИНТУИЦИОНИСТСКОГО ИСЧИСЛЕНИЯ ВЫСКАЗЫВАНИЙ

Ляхов А.В., к.т.н., Кучер В.А., к.э.н.
Донецкий национальный технический университет,
Донецк, Украина

Логические грамматики превратились с течением лет в инструментарий высокого уровня, и теперь они позволяют пользователю сконцентрироваться на лингвистических феноменах. Грамматики, построенные на определенных предложениях, поддерживают использование логики для обработки данных естественного языка, и они подготовили почву для практической работы лингвистов на языке программирования PROLOG.

Грамматика, построенная на определенных предложениях (DEFINITE-CLAUSE GRAMMAR или DCG)

Проведение работ по исследованию построения грамматик для понимания естественного языка (далее ЕЯ) приобрело большую популярность после введения Колмеройером в 1975 году грамматических формализмов, основанных на предложениях Хорна. Так называемые метаморфозные грамматики (metamorphosis grammars или MGs) определили рост заинтересованности в области выражения лингвистических понятий в логике (qv) и заложили основу для создания качественных блоков предварительной обработки данных (процессоров ввода-вывода) и интерфейсов. Основные результаты этих исследовательских работ нашли в консультировании и создании баз данных на основе ЕЯ, выработке вопросов и ответов, переводе текстов и синтезе текстов, исходя из надлежащим образом оформленных спецификаций [1; 2].

Понятие грамматик, построенных на определенных предложениях (DCGs), как особого случая метаморфозных грамматик, было введено в 1978 году Перейрой и Уорреном в качестве грамматического формализма, для которого PROLOG имеет эффективный механизм синтаксического анализа. Одни практические системы были созданы для одновременного использования синтаксического и семантического знания для привнесения логики в структуру, содержащую в себе информацию для семантической интерпретации. Другие системы были выстроены на более чем одном уровне трансляции; использование синтаксического и семантического знания осуществлялось отдельно друг от друга, и конечным результатом являлось в PROLOGe предложение Хорна, выполнение которого осуществлялось механизмом планирования (qv).

Техника экстрапозиционных грамматик (extraposition grammars или XGs) была предложена Перейрой для описания определенных глобальных отношений или экстрапозиций, таких как связь между относительным местоимением и его записью. В конечном итоге разработки наподобие грамматик структуры определений (modifier structure grammars или MSGs) Даля и МакКорда, древовидных грамматик (tree grammars или TGs) Колмеройера и усложненных грамматик (puzzle grammars или PGs) Сабатье увеличили силу выражения лингвистических понятий.

Все эти исследовательские работы по грамматическим формализмам, замешанным на логике, стали возможны и проще в осуществлении после выбора PROLOGa, языка программирования основанного на подмножестве логики первого порядка [3].

ЛОГИЧЕСКИЕ ГРАММАТИКИ

Граматики описывают структуру (синтаксис) языков множеством продукций (правил, перерабатывающих текст). Например, правилом

sentence -> noun-phrase verb-phrase

устанавливается связь между тремя нетерминальными символами: предложение может состоять из именной группы и следующей за ней глагольной группы.

Такие правила могут быть отображены в PROLOGe следующим образом:

sentence (S1, S3): - noun-phrase (S1, S2), verb-phrase (S2, S3).

Verb-phrase (S1, S2): - connects (S1, writes, S2).

Connects (1, each, 2).

Connects (2, author, 3).

Connects (3, writes, 4).

(Примечание: предикаты (т. е. выражения с неопределенными терминами, или переменными, которые преобразуются в истинные или ложные высказывания при выборе конкретных значений для этих самых терминов) заносятся в PROLOG через запятую. Переменные отличаются от констант первой заглавной буквой).

В нижеследующей записи числа обозначают начало и конец каждого слова:

$_1\text{each}_2\ \text{author}_3\ \text{writes}_4$

Чтобы проверить правильность построения предложения, необходимо указать цель

? – sentence (1, 4).

(где ? – бинарное обозначение структуры (или бинарный функтор), содержащееся в любой системе PROLOG) и продемонстрировать, что она подтверждается предыдущими условиями. Используя список в качестве информационной структуры для представления предложения, числа больше не нужны, так как PROLOG имеет устройство синтаксического анализа, способного перевести:

? – sentence ([each, author, writes]. []).

Граматики, построенные на определенных предложениях, являются объемом понятия контекстно-свободных грамматик, которые также могут быть транслированы на язык PROLOG. Граматики, построенные на определенных предложениях, позволяют любому логическому выражению стать нетерминальным, они построены на логических символах: константах, переменных, выражениях, - а не только на одних константах [4; 5]. Также они имеют только один нетерминальный символ в левой части каждого правила. Контекстные зависимости (контекстные отношения подчинения) описываются логическими переменными в рамках параметров (или независимых переменных) грамматических символов.

У правила грамматики, построенной на определенных предложениях, следующая форма:
nonterminal symbol -> body (основная часть программы).

Где “body” (“основная часть”) является последовательностью одного или более элементов данных, отделенных друг от друга запятыми. Каждый элемент данных является либо нетерминальным символом, либо последовательностью терминальных символов. Значение правила состоит в том, что “основная часть” - это возможная форма для группы типа “нетерминальный символ”. В PROLOGе нетерминальный символ записывается как выражение (а не как список), а последовательность терминальных символов – в виде списка.

В правой части правила наряду с нетерминальными символами и списком терминальных символов могут находиться последовательности вызовов процедур, записываемых в фигурных скобках ({and}). Они используются для выражения дополнительных условий, которые в обязательном порядке должны выполняться, чтобы правило действовало. Нетерминальный символ преобразуется в $(N + 2)$ -местный предикат (имеющий идентичное название), чьи первые N параметры полностью описаны в нетерминальном символе и чьи последние два параметра являются такими же, как и при трансляции контекстно-свободного нетерминального символа. Вызовы процедур в правой части правила транслируются так, как они есть.

Каждое грамматическое правило, типа

$p(X) \rightarrow q(X)$.

Получает группу входящих данных, анализирует некую исходную часть и генерирует остаток для дальнейшего анализа. Это частное правило транслируется системой PROLOG как

$p(X, S_0, S) : - q(X, S_0, S)$.

Следовательно, система грамматической индексации в PROLOGe обеспечивает более сжатую запись, когда параметры для групп входящих и выходящих данных неявно выражены.

Когда в правиле содержатся терминальные символы, они транслируются со сказуемым "connects". К примеру,

connects(S1, X, S2)

означает, что суть S1 связана через X с сутью S2.

Правило

$p(X) \rightarrow (\text{older}), q(X), (\text{high}).$

Преобразуется в

$p(X, S0, S): - \text{connects}(S0, \text{older}, S1),$
 $q(X, S1, S2),$
 $\text{connects}(S2, \text{high}, S).$

АНАЛИЗ ЕЯ

В 1977 году Колмеройер ввел общую схему анализа ЕЯ, что явилось решающим шагом вперед и вызвало огромный интерес в области использования логических грамматик как альтернативы прочно утвердившимся грамматикам расширенных сетей переходов (augmented transition network grammars). С исторической точки зрения, это можно признать поворотным пунктом, так как был указан метод преобразования предложений ЕЯ в логические структуры. Метод заключался в рассмотрении простейших высказываний, содержащих собственные имена существительные, какой бы то ни было артикль в виде квантора (кванторного слова) с тремя операциями перехода и четыре приоритетных правила, для разрешения проблемы иерархии квантования (количественного измерения качественных признаков).

Краткий обзор общей схемы побуждает к дальнейшему развитию приемов составления логических грамматик. Например, предложение

Chomsky is (a) writer

содержит имя собственное, и глагол "to be" преобразуется в формулу
writer(chomsky)

Главным образом глаголы, прилагательные, имена существительные вводят характеристики с n параметрами. Для глаголов, n может равняться 1 [непереходные глаголы] или $N+1$ [переходные глаголы, где N – количество комплементов (дополнений, следующих за глаголом в составном сказуемом и описывающих его подлежащее)]. Для прилагательных и имен существительных n равняется или больше 1 (отношения, где n – это n -место его параметров). Параметры представляют в предложении дополнения при имени существительном, глаголе или прилагательном.

Например, предложение

Chomsky writes a book

содержит глагол "write", существительное "book" и артикль "a" и может быть заменено следующей формой:

for a
B
such that
B is (a) book (1)
it is true that
Chomsky writes B (2)

где (1) и (2) являются простейшими высказываниями.

Эта же логическая структура может быть записана в стенографических индексах:

$a(B, \text{book}(B), \text{writes}(\text{Chomsky}), B)).$

Обратите внимание, что высказывания (1) и (2) преобразуются в формулы “book(B)” и “writes(Chomsky, B),” соответственно.

Логическая структура выражает смысл предложения, и каждая из ее составных частей соответствует смыслу обособленных слов согласно принципу Фреге. Записи, передающие значение, указываются как логические структуры, так как единственным из области значения, что люди научились точным образом представлять в виде записи в ЭВМ, являются логические отношения.

Всякий артикль a представляет квантор q (кванторное слово) с тремя операциями перехода, создающий новую формулу из переменной x и двух формул $f1$ и $f2$,

$$q(x, f1, f2).$$

Эта формула соответствует высказыванию

for a x such that $e1$, it is true that $e2$

где $e1$ и $e2$ являются простейшими высказываниями соответствующими $f1$ и $f2$.

Например, предложение

Chomsky writes a book for each publisher

содержит глагол “write”, два существительных (“book” и “publisher”) и два артикля (“a” и “each”) и может быть заменено следующей формой:

for each P such that P is a publisher it is true that for a B such that B is a book, it is true that

Chomsky writes B for P

Предложение преобразуется в логическую структуру

each(P ,
publisher(P),
a(B ,
book(B),
writes-for(Chomsky, B , P))).

Эта логическая структура отражает следующее приоритетное правило: в конструкции с использованием имени существительного (book) и его дополнения (publisher), квантование, представленное артиклем при дополнении, влияет на квантование, представленное артиклем при имени существительном. Помимо этого правила Колмеройер предложил еще 3 приоритетных правила для организации возможности квантования.

АНАЛИЗ КОЛМЕРОЙЕРА ДЛЯ КОНКРЕТНОГО ЕЯ

Общая схема анализа была изначально предложена Колмеройером для французского и английского языков. Позднее Даль адаптировал ее для испанского, Коэлхо – для португальского, а Пик предложил иную семантику для артиклей французского языка.

ОБЩАЯ СХЕМА АНАЛИЗА КОЛМЕРОЙЕРА В КАЧЕСТВЕ ГРАММАТИКИ, ПОСТРОЕННОЙ НА ОПРЕДЕЛЕННЫХ ПРЕДЛОЖЕНИЯХ (DCG)

DCGs поддерживают процессы синтаксического анализа и транслирования благодаря взятым ими на вооружение из подмножества ЕЯ, необходимым элементам синтаксиса и семантики. Процесс синтаксического анализа состоит из доказательства, что цепочка слов является законным и правильно построенным предложением (с точки зрения выбранного синтаксиса). Процедура доказательства осуществляется через исследовательскую стратегию (с конца на начало, сверху вниз, слева направо) и через правило умозаключения (логического вывода), где заключением является qv , уже после отработки системы PROLOG. Трансляция заключается в изображении каждого предложения в виде логической структуры. Эта структура состоит из правильно построенных формул определенной логической системы, которая основывается на объеме понятия логики предикатов(qv).

Механизм транслирования выражается как множество определенных предложений логики через грамматические правила PROLOGa. Он (механизм) может содержать, как вместе, так и

по отдельности синтаксические и семантические знания из подмножества рассматриваемого ЕЯ. Механизм синтаксического анализа зависит от системы PROLOG, и он может быть вскрыт включением отслеживающего устройства. Трансляция и синтаксический анализ – независимые друг от друга процессы, их независимость позволяет легче производить изменения в грамматике.

Упрощенная грамматика, обозначаемая через G, рассматривается следующим образом. Она производит синтаксический анализ английских предложений, одновременно порождая соответствующие им логические структуры. Грамматика определяется 2 модулями:

1) синтаксис + семантика

2) морфология

и имеет дело с предложениями типа: Hodges writes for Penguin.

СИНТАКСИС + СЕМАНТИКА

sentences(S) -> noun-phrase(NP, S2, O),
verb([subject-X | L], O1),
complements(L, O1, O2).

Complements([], O, O) -> [].

Complements([K-N | L], O1, O3) -> complements(L, O1, O2),
case (K),
noun-phrase(N, O2, O3).

Noun-phrase(N, O2, O4) -> article(N, O1, O2, O3),
common-noun([subject-N | L], O1),
complements(L, O3, O4).

Noun-phrase(PN, O, O) -> [PN], {proper-noun(PN)}.

Article[A, O1, O2 and (O1, O2)] -> [a].

Case(for) -> [for].

Case(direct) -> [].

МОРФОЛОГИЯ

verb([subject-A, for -P], is-published-by(A,P)) -> [writes].

Common-noun([subject-P], publisher(P)) -> [publisher].

Proper-noun(hodges).

Proper-noun(penguin).

К примеру, правило:

noun-phrase(PN, O, O) -> [PN], {proper-noun(PN)}.

Представляет собой структуру:

noun-phrase(PN, O, O, S0, S): - connects(S0, PN, S),
proper-noun(PN).

Первое правило грамматики G, по идее, признается справедливым только для предложений, в которых за именной группой следует глагол с возможно некоторыми дополнениями. Первое грамматическое правило для дополнений допускает их отсутствие (терминальный символ [] выступает в роли пустого списка), второе правило для дополнений определяет последовательность дополнений как группу данных, составленную из дополнения, падежа и именной группы.

Различные параметры отличных друг от друга нетерминальных символов связаны одной и той же логической переменной. Это позволяет строить структуры в процессе унификации.

Именная группа “a publisher” анализируется и транслируется грамматическим правилом как:

noun-phrase(N, Oa, Ob) -> article(N, Oc, Od, Oe),
common-noun(N, of),
{constraints(Oa, Ob, Oc, Od, Oe, Of)}.

Заметьте, что это правило – упрощенная версия четвертого правила представленной грамматики G. Нетерминальный символ для именной группы имеет 3 параметра. Интерпретация последнего параметра Ob будет зависеть от характеристики Oa индивидуума N, так как в общем именная группа содержит артикль, такой как “a”.

Теперь смотри на правую часть правила. Слово “a” имеет интерпретацию Oe,
and(Oc, Od)

в контексте двух характеристик Oc и Od индивидуума N. Характеристика Oc будет соответствовать оставшейся части именной группы, содержащей слово “a”, а характеристика Od вытекает из остатка предложения. Значит, Oe будет содержать всеобъемлющую интерпретацию и связываться с Ob через одну и ту же переменную. Так как Of является характеристикой нарицательного существительного, она связывается с Oc одной и той же переменной. Oa имеет описание характеристик N, а также зависит от характеристик оставшейся части предложения. Поэтому Oa связывается с Od через одну и ту же переменную.

Каждое слово ассоциируется с характеристикой. Например, значение глагола “writes” вводится отношением “is-published-by(A,P)”. Глагольное правило также содержит информацию о характеристиках отношения, а именно то, что “A” играет роль подлежащего предложения а “P” “навязывает” использование предлога “for”. Значение неопределенного артикля “a” вводится конъюнкцией “and(O1, O2)” согласно определению, принятому в классической логике.

Более продвинутой по сравнению с G грамматика имела бы более скрупулезно разработанные дефиниции существительных, глаголов, прилагательных, артиклей:

noun([A-[] & author & type-X], pr(author(X))) -> no(author, A).

No(Type, GN) -> [Noun], {no1(Noun, Type, GN)}.

No1(author, author, mas-sin).

Verb([(G-N)-V&type-X, dir-A-W&title-Y], pr(author(X, Y))) -> ve(writes, N).

Ve(Type, N) -> [Verb], {ve1(Verb, Type, N)}.

Ve1(writes, writes, sin).

Adjective([A-{ } & author & type-X, prep(by)-_[] & pub & type-Y], pr(published(Y, X))) -> ad(pub, A).

Ad(Type, GN) -> [Adj], {ad1(Adj, Type, GN)}.

Ad1(published, pub, mas-sin).

Article(G-sin)-D-X, O1, O2, for([X, D] and (O1, O2)), cardinality(X, greater, 0))) art-ind(G-sin).

Art-ind(mas-sin) -> [a]; [some].

(Замечание: безымянные переменные записываются в PROLOG как “_”.)

Эти дефиниции включают синтаксические и семантические проверки, такие как грамматический род, число, семантические типы. Значение артикля также отличается. Вместо квантора с двумя операциями перехода оно было введено квантором с тремя операциями перехода. Первая операция перехода – квантование переменной X, вторая – для основной характеристики “and” переменной X, третья – для точного определения характеристики (мощности множества) и ограничения области переменной X.

ОБЛАСТЬ ПОНЯТИЙ ГРАММАТИК, ПОСТРОЕННЫХ НА ОПРЕДЕЛЕННЫХ ПРЕДЛОЖЕНИЯХ

Экстрапозиционные грамматики (XGs) увеличивают мощь DCGs при перечислении контекстных зависимостей [6; 7]. Правила XG могут иметь в своей левой части более одного нетерминального символа и символ пробела “ ”, выражающий случайную цепочку логических символов (терминальных и нетерминальных). Например, правило экстрапозиционной грамматики

Relative-marker . . . complement -> [that].

Утверждает, что относительное местоимение "that" может быть проанализировано как относительный показатель, за которым следуют какие-нибудь неизвестные фразы и затем дополнение.

XGs упрощают выражение синтаксических представлений и следовательно позволяют упростить трактовку семантических и логических описаний. Параметры для нетерминальных символов используются (как и в DCGs) для проверок согласования, производства дерева синтаксического анализа и ограничения возможности присоединения постмодификаторов.

Граматики структуры определений (MSGs) увеличивают вероятность точного описания несинтаксических репрезентаций. Они упрощают автоматическое моделирование таких репрезентаций при одновременно происходящем анализе.

Древовидные (или древесные) грамматики (TGs) позволяют лучше ориентировать лингвистические конструкции.

Усложненные грамматики (PGs) являются средством, разработанным специально для нужд лингвистов. Правила их стратегии описывают порядок и режим трансляции, и описываются эти правила независимо друг от друга.

Литература

1. Грис Д. Наука программирования: Пер с английского – М.: Мир, 1984. – 416 с.
2. Клини С.К. Введение в математику. М.: изд-во иностранной литературы, 1975. – 525 с.
3. Непейвода П.П. Соотношение между правилами естественного вывода и операторами алгоритмических языков высокого уровня. М.: Изд-во ДАН СССР, 1978, т. 239, № 3. – 4 с.
4. Непейвода П.П. О построении правильных программ. Вопросы кибернетики, вып. 46. – 43 с.
5. Непейвода П.П. Об одном методе построения правильной программы из правильных подпрограмм. Программирование, 1979, – № 1. – 11 с.
6. Правиц Д. Натуральный вывод. М.: ЛОРИ, 1977. – 219 с.
7. Бельтюков А.П. Иерархии и машинное описание малых субрекурсивных классов // Известия института математики и информатики УдГУ. – Ижевск. – 2002. – № 2. – С. 3 - 89.