**Искусственный интеллект.**

Принцип аналогии в морфологии

В работе Белоногова Г. Г. и Зеленкова Ю. Г. описывается принцип построения алгоритма морфологического анализа текстов на основе принципа аналогии. Данный принцип используется в системах орфографического контроля русских текстов, системах автоматического индексирования документов и системах машинного перевода текстов с русского языка на английский и с английского языка на русский. Производительность программы на компьютере с процессором от 386 и выше составляет около 400 слов/с.

При автоматической обработке текста возникает проблема “новых“ слов. Для синтаксического анализа и синтеза необходимо знать грамматические характеристики слов. Если слова в словаре нет, то морфологический анализ не может быть выполнен, а следовательно не могут быть определены грамматические характеристики слова.

Для того, чтобы определить грамматические характеристики слов без словаря, Белоногов предложил принцип аналогии. Он основан на том, что существует сильная корреляционная связь между грамматическими характеристиками слов и буквенным составом их концов. Например: организация, приватизация, концентрация имеют ж. р., им. п. и ед. ч.; работают, понимают, привлекают - это глаголы в 3-ем лице мн. ч. и т. д.

Принцип аналогии проверялся на ряде индоевропейских языков: (русский, болгарский, латышский, испанский, английский) и оказался эффективным. Сначала он применялся для определения грамматических характеристик слов, не включенных в машинный словарь. Затем возникла идея при проведении морфологического анализа отказаться от машинного словаря.

Если по текстам большого объема составить словарь словоформ и назначить каждой словоформе некоторые грамматические признаки, а затем преобразовать данный словарь в обратный словарь словоформ, то можно обнаружить, что многие участки словаря имеют одинаковые наборы признаков.

Обратный словарь словоформ представляет собой список словоформ с такими характеристиками как признак длинны грамматического окончания, номер флективного класса (типа словоизменения) и числовой индекс, характеризующий такие признаки как “глагольность“, “местоименность”, “сравнительная степень”. Например:

масштаба 01/001/01

служба 01/056/01

возникшие 02/105/10

батальон 00/021/01

рассчитывая 00/152/10

Обратный словарь используется для автоматического морфологического анализа текстов, если составляющие их словоформы отождествлять со словоформами словаря и приписывать им грамматическую информацию, указанную в словаре. Словоформам текста, которые не находятся в словаре, можно приписывать грамматическую информацию тех словоформ словаря, концы которых в максимальной степени совпадают с концами этих новых словоформ текста.

Объем обратного словаря можно сократить, если на всех его участках оставить по две словоформы: начальную и конечную. Более того из этих двух словоформ можно оставить только одну, и если словоформа текста не совпадет ни с одной словоформой обратного словаря, то ей приписывается информация непосредственно предшествующей словоформы этого словаря.

Данный сокращенный словарь можно еще сократить, если исключить из него начальные буквы словоформ, не оказывающие влияние на результаты морфологического анализа. При этом у каждой пары рядом стоящих словоформ оставляются справа совпадающие конечные буквосочетания и еще по одной букве, которые не совпадают. Например:

аба 01/001/01

еба 01/044/01

неба 01/071/01

авшие 02/105/10

 тальон 00/021/01

 тывая 00/152/10

После выполнения всех операций объем словаря сокращается в 8 раз. На точность первоначально включенных в словарь словоформ это не повлияет, а точность анализа остальных словоформ русского языка будет достаточно высокой.

Для морфологического анализа текстов на основе метода аналогии достаточно располагать обратным словарем концов слов. Но авторы разработки сделали еще “Словарь служебных и коротких слов”. В этот словарь были включены сначала предлоги, местоимения, частицы, союзы и короткие слова до 5 букв. Затем в него вошли также словоформы, которые по методу аналогии анализировались неверно. В результате этот словарь увеличился до 11 тысяч словоформ.

Таким образом, в процессе морфологического анализа словоформы ищутся в словаре “Служебных и коротких слов”, а затем в словаре концов словоформ. Результаты анализа, полученные по первому словарю, считаются более надежными, и словоформы, найденные в этом словаре, дальнейшей обработке не подвергаются.

В настоящее время вероятность правильного анализа слов при обработке текстов любой тематики превышает 99%.

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. Основное применение 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.

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, содержа 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 PROLOGе 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 PROLOGа, языка программирования основанного на подмножестве логики первого порядка.

ЛОГИЧЕСКИЕ ГРАММАТИКИ

Грамматики 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 PROLOGе 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:

1each2 author3 writes4

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]. [ ]).

Грамматики, построенные на определенных предложениях, являются объемом понятия контекстно-свободных грамматик, которые также могут быть транслированы на язык PROLOG. Грамматики, построенные на определенных предложениях, позволяют любому логическому выражению стать нетерминальным, они построены на логических символах: константах, переменных, выражениях, - а не только на одних константах. 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 выполнятся, 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) -> 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, S0, S): - q(X, S0, S).

Следовательно, система грамматической индексации в PROLOGе 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) -> (older), q(X), (high).

преобразуется в

|  |  |
| --- | --- |
| p(X, S0, S): -  | connects(S0, older, S1), |
|  | q(X, S1, S2), |
|  | connects(S2, 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, book (B), writes(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 Фреже. 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 PROLOGа. 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:

синтаксис + семантика

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 продвинутая 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. 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.

Грамматики 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.

Порождение текстов на естественном языке

Порождение текстов на естественном языке - процесс преднамеренного построения текста на естественном языке с целью решать определенные коммуникативные задачи. Термин "текст" рассматривается как общий, рекурсивный термин, который может относится к письменному или устному высказыванию, или к отдельным частям высказывания. При порождении текстов, в устной или письменной форме, человеку важно обдумать и отредактировать производимое высказывание. Едва ли можно сказать, что большинство программ может “говорить” сегодня, в основном все они лишь выводят слова на экран. Так как для программы порождения текстов на сегодняшний день не стоит вопрос конструирования фразы, эти детали принимаются во внимание только тогда, когда они задействованы в создании программы.

Цели исходят из другой программы, возможно экспертной рассуждающей системы или ICAI обучающей программы, которая общается с пользователем на естественном языке. Произведенные тексты могут быть различной длины: от одиночной фразы, данной в ответ на вопрос, до диалогов с большим количеством предложений или толкований на целую страницу. Порождение текстов на естественном языке отличается от программ, просто использующих естественный язык. Программы, печатающие сообщения на естественном языке, существуют со времен появления компьютеров, но сейчас, например, никто не хочет разбираться, каким образом построены сообщения об ошибках при компиляции на ФОРТРАНе, как бы правильно они не были написаны. Сообщение об ошибках ничего не "означает" для программы, которая печатает их: связь между цепочкой слов и работой программы создается программистом. Даже использование утверждений с параметром, где зафиксированная цепочка слов может быть увеличена именами или простыми описаниями, заменяющими переменные, не является собственно порождением текстов на естественном языке. Успех таких приемов как “заполнить пробелы” или “шаблон” зависит от количества и сложности ситуаций, в которых программа должна использовать их. То, что они были адекватны до сих пор для работы программы, объясняется, по большей части, относительной простотой сегодняшних программ, чем возможностями порождения с использованием метода “шаблона”.

В отличие от таких "инженерных разработок", исследование порождения текстов на естественном языке, подобно другим областям вычислительной лингвистики (qv), имеет своей целью компьютерное моделирование человеческой способности к порождению высказываний. Основное внимание при этом сосредотачивается на объяснении двух ключевых вопросов: многосторонность и творческий потенциал. Что люди знают относительно их языка, какие процессы они при этом используют, что дает возможность им быть универсальным, изменяя тексты в форме и акцентировании, чтобы покрыть огромный диапазон языковых ситуаций?

В этой статье описываетcя исследование в области ИИ по порождению естественных языков, при этом особое внимание уделяется конкретным проблемам, которые требуют разрешения. Статья начинается с противопоставления порождения пониманию, чтобы установить базисные понятия разложения процесса на компоненты. Далее приводятся примеры, показывающие работу некоторых порождающих систем, их возможности и трудности, с которыми они сталкиваются.

В оставшейся части статьи рассматриваются общие подходы к порождению речи, включая характерные описания порождающего словаря. Отдельный раздел продолжает обзор альтернативных подходов к представлению и использованию грамматики.

Характер процесса порождения. В отличие от организации процесса понимания, который, на первый взгляд, может следовать традиционным стадиям лингвистического анализа: морфология, синтаксис, семантика, прагматика /дискурс¦ процесс порождения имеет существенно отличный характер. Этот факт следует непосредственно из присущих различий в информационном потоке в двух процессах. Понимание осуществляется от формы к содержанию; порождение есть совершенно противоположный процесс. При понимании, формулировка текста (и, возможно, интонация) - "известны". Из формулировки процесс создает и выводит примерное содержание, переданное текстом и, вероятно, усилиями диктора в создании текста. Первым делом следует просмотреть слова текста последовательно, в течение чего форма текста постепенно разворачивается. Главные проблемы вызваны неоднозначностью¦ одна форма может содержать диапазон альтернативных значений, и аудитория получает большее количество информации из ситуационных заключений, чем это может быть фактически передано текстом. Кроме того, несоответствия у диктора и аудитории модели ситуации ведут к непредсказуемым заключениям.

Порождение имеет противоположный информационный поток. Оно переходит от содержания к форме, от целей и перспектив к линейно упорядоченным словам и синтаксическим маркерам. Модель ситуации и дискурс обеспечивают основу для создания выбора среди альтернативных формулировок и конструкций, которые производит язык: первое в построении заранее обдуманного текста. Большинство систем порождения производит поверхностные тексты последовательно слева направо, но только приняв решение сверху-вниз по содержанию и форме текста в целом. Проблема генератора состоит в том, чтобы выбрать из поставленных источников, как правильно сообщить о желаемых умозаключениях аудитории и какую информацию опустить из явного упоминания в тексте.

Можно вообразить, что процесс порождение также организован, как и процесс понимания, только в противоположном порядке. К некотором смысле это верно: идентификация намерения (цели) в значительной степени предшествует любой детализации информация, которая предназначается для аудитории: планирование риторической структуры, например, в значительной степени, предшествует любой синтаксической структуре, а синтаксический контекст слова должен быть зафиксирован, прежде чем будут известны морфологическая и суперсегментная формы, которые примет слово.

Синтаксис и словарь языка становится как ресурсами, так и ограничениями, определяя элементы, доступные для создания текста, а также зависимости между ними, которые определяют возможные правильные комбинации. Эти зависимости, и тот факт, что они по умолчанию управляют, когда информация, от которой зависит каждое решение, становится доступной, - основная причина, почему программы порождения в значительной степени следуют стандартным стадиям, определенными лингвистами. Идентификация цели предшествует выбору содержания и риторическому планированию, которое предшествует синтаксической конструкции, только потому что это - естественный порядок принятия решения; проще следовать потоку зависимостей, чем перепрыгивать и принимать случайное решение, которое может оказаться преждевременным и несостоятельным. Сегодняшнее исследование сосредоточено как на понимании, как лучше представить решения, которые являются возможными, и зависимости среди них, так и на том, как представить ограничения и возможности раньше решений, которые встанут на место последних во время процесса порождения.

Стандартные Компоненты и Терминология. Компоненты порождения естественного языка не существуют сами по себе. Они расположены внутри человеко-машинного интерфейса, который также используют и компоненты понимания естественного языка, - ВВОД в систему. В хорошем человеко-машинном интерфейсе сегодня также хотелось бы видеть координированную графическую поддержку ввода и вывода, дополняя систему ВВОДа-ВЫВОДа естественного языка. Интерфейс может закончиться здесь, а может также включать в себя другие общедоступные компоненты, типа контроллера дискурса, который указывает генератору, какие действия нужно предпринять, а также координирует интерпретации, сделанные компонентом понимания. За интерфейсом следует ­нелингвистическое рассуждение (qv) или программа базы данных, которую пользователи используют в качестве речевого интерфейса. Эта программа будет упоминаться в этой статье как основная программа; ею может оказаться любая система ИИ: совместная база данных, экспертная диагностическая система, ICAI обучающая программа, комментатор, программа-консультант, машинный переводчик. Тип основной программы теперь не имеет никакого значения для самой порождающей системы (генератора естественного языка).

Сегодня большинство исследователей в этой области работает, в основном, с экспертными системами, где процесс общения контролируется программой, а не пользователем. Кроме того, ЭС и интеллектуальные машинные обучающие программы, вероятно, способны понимать довольно сложные тексты, что делает их привлекательными для специалистов, готовых работать с уже разработанными системами.

Процесс порождения начинается внутри основной программы, в случае, когда, например, необходимо ответить на вопрос пользователя; или во время беседы может возникнуть потребность прервать действия пользователя, чтобы указать надвигающуюся проблему. Как только процесс инициализирован, три вида действий должны быть выполнены:

1. Идентификация целей высказывания,

2. Планирование, как эти цели могут быть достигнуты, включая оценку ситуации и доступных коммуникативных ресурсов,

3. Реализация планов в текст.

Цели должны обычно передавать некоторую информацию аудитории или побуждать их к действиям или рассуждениям. Социальные и психологические, а также практические мотивы, побуждающие человека к общению, естественно, неприменимы для сегодняшних компьютерных программ. Планирование включает в себя отбор (преднамеренное вычеркивание) информационных модулей, которые появляются в тексте (например, концепции, отношения, индивидуальность).

Реализация зависит от знания грамматики языка и правил связности дискурса, и дает синтаксическое описание текста как промежуточное представление. При этом выделяется не только лингвистическая форма, но также знание относительно критериев, которые показывают, как используются эти формы. В многих исследованиях процесс, который проводит грамматическую реализацию, называется лингвистическим компонентом(10), а иногда планирование и вместе с процессом идентификации цели называется стратегическим компонентом (13). Обычно это - только лингвистический компонент, который имеет любое прямое знание относительно грамматики производимого языка. Какую форму эта грамматика принимает - один из самых больших различий среди проектов порождения.

Традиционно для лингвиста, грамматика - костяк в отрезке утверждения/ высказывания. Содержание утверждений - специфические факты данного естественного языка - не представляет такого интереса для лингвиста.

Аналогичная ситуация с порождением текстов, за исключением того, что запись - процедурная и декларативная - разработана, чтобы обеспечивать очень специфическую функцию, с которой традиционный лингвист не сталкивается, а именно: вести и сдерживать процесс порождения текста со специфическим содержанием и целями в присутствии специфической аудитории. Грамматика теперь ответственна за наличие выбора, который язык предоставляет для формы и словаря. Исследователи порождения должны сделать верный выбор, чтобы, используя функции различных конструкций для достижения конкретной цели. Другая функция грамматики - следить за грамматичностью текста, т. е. определение зависимостей и ограничивая решения.

Технический уровень

Разноплановое развитие и творческий потенциал в порождении текстов является возможным при следующих условиях:

1. Генератор включает в себя весь объем основной грамматики;

2. Основная программа имеет сложное, разносторонее, концептуальное представление(вид);

3. Текстовый планировщик может использовать модели аудитории и дискурса.

К сожалению, такие генераторы - все еще только предмет исследования сегодня, т. к. техническая сторона остается на уровне программы SHRDLU Винограда в 1970 (17), которая порождала предложения в процессе ответа на вопросы, система “непосредственной замены”, порождающая простые грамматические глагольные корректировки в целях достижения удобочитаемого текста.

When did you pick up [the green pyramid]?

While I was stacking up yhe red cube, a large red block, and a large green cube.

К концу 1970-ых такие системы стали достаточно популярны в работе ЭС: для перевода многочисленных правил в этих системах. Необходимость программ порождения текстов в системах с составной структурой и коммуникативным контекстом была очевидной.

Исследователи заинтересованы в более сложных текстах, нежели в контекстно-свободных представлениях, которые требуются правилами системы. В качестве примера приводится простое описание из программы Сигурда, чья цель была выяснить, как в помощью интонации выявляется группировка:

The submarine is to the south of the port. It is approaching the port, but is not close to it. The destroyer is approaching the port too.

Использование слов-ссылок “but” “too” является большим прогрессом в структурировании системы. Предложение, которое является источником в базе данных ЭС , рассуждающее о субмаринах и эсминцах, не будет обрамлено концептуальными эквивалентами таких функциональных слов, и может быть прочтено простым шаблоном, потому что ссылки специфичны и могут быть употреблены только в отдельном конкретном случае.

Еще одна техническая, пока не разрешенная, проблема - “последующая ссылка”. Какими должны быть слова-заменители, если предмет появляется больше, чем один раз в тексте? Постоянное употребление местоимений может привести к неоднозначности. В качестве примера приводится отрывок из исследований Гранвилле, который классифицирует отношения между референтом и предметом и разрабатывает правила, по которым бы могли строиться последующие ссылки.

Pogo cares for Hepzibah. Churchy likes her, too. Pogo gives a rose to her, which pleases her. She does not want Churchy’s rose. He is jealous. He punches Pogo. He gives a rose to Hebzibah. The petals drop off. This upsets her. She cries.

Неудивительно, что у исследователей, разрабатывающих основную программу, генераторы обладают наибольшей эффективностью, что дает уверенность в том, что имеется концептуальная основа для группирования отдельных предложений/ утверждений в тексте. Важным моментом на этом этапе является программа PROTEUS, разработанная Дэйви в 1974. Программа дает описание игры крестики-нолики и считается одной из программ, наиболее свободно владеющей естественным языком. PROTEUS имеет модель толкования конкретных шагов: нападение, встречное нападение, включает в себя риторический принцип, что в текст нужно помещать только наиболее существенную информацию в ситуации. Грамматика и средства реализации выбирают описанные и сгруппированные шаги, исправляют формы, так чтобы они были грамматичны в английских предложениях, и порождают собственно текст.

Следует упомянуть и программу ERMA Клиппенгера (1974)- единственная программа на тот момент, работающая со спонтанной речью. Как люди размышляют о том, что они говорят, как они динамически планируют или меняют свои намерения относительно того, что они хотят сказать в разговоре? В целях моделирования этого процесса, Клиппенгер анализировал стенограмму речи пациента по психоанализу с тем, чтобы понять рассуждения пациента, дающие объяснение одному из параграфов стенограммы, который ERMA могла подробно воспроизвести. Клиппенгер разработал структуру из пяти основных взаимосвязанных компонентов, участвующих в порождении спонтанного текста. Но для компьютерного программирования в 1974 реализовать этот план было не под силу, вследствие чего проект был оставлен.

Исторический обзор проблемы. По сути дела, программы PROTEUS Дэйви и ERMA Клиппенгера являются самыми старшими в этой области. Во-первых, потому что до начала 80-ых сравнительно мало людей работало над проблемой порождения , во-вторых, сама проблема достаточно сложна, по мнению авторов статьи, намного сложнее проблемы понимания речи. На самом деле, проблемой серьезно занимались в начале 1970-ых. Но справедливо отметить, что на важной конференции по данной проблеме в 1975г представленные отчеты о проделанной работе не нашли должного отклика, после чего исследования по порождению естественного языка были почти приостановлены до начала 1980-ых.

До 80-ых специалисты в области ИИ склонны были считать проблему порождения достаточно легкой. В самом деле, разве трудно взять к-л утверждение из некоторого речевого фрагмента, связать его с определениями, хранящимися отдельно, и произвести, например, следующее “The big black block supports a green one”. Это было под силу SHRDLU Винограда уже в 1970г. Если бы можно было ограничиться этими знаниями, то, на самом деле, не возникало бы проблем. Но вариативность языка не давала такой возможности. Каким образом человек представляет грамматические знания, которые позволяют генератору использовать синтаксическую структуру предложения в целях cоздания соответствующего относительного предложения (“the green block that’s supported by the big red one”, “a green one”, а не “a green block”), а также вообще иметь представление о возможности таких относительных предложений и подобных замен.

Общие подходы к проблеме. Трудно идентифицировать общие элементы в различных проектах исследования по порождению естественного языка. Напротив, в исследованиях по пониманию речи можно выделить несколько основных подходов к проблеме: использование расширенных сетей переходов, семантические грамматики (qv), рабочие системы, основанные на представлении концептуальной зависимости, процедурная семантика и многое другое. Исследование порождения не может дать подобной классификации, поскольку очень мало специалистов ставили эту проблему во главу угла. Большие исследовательские группы, полностью сконцентрировавшиеся на вопросе порождения естественного языка, начали создаваться в последние два года. Основная проблема состоит в отсутствии общего отправного пункта, конкретной основы для сравнения, что осложняет работу, не дает возможности для взаимопомощи между исследователями: практически невозможно проверить свои эксперименты на системе другого разработчика. Однако имеются общие нити, связывающие различные проекты: похожие подходы, похожие представления, похожие грамматики.

Существует два вопроса, представляющих общий интерес. Первый вопрос: как сопоставить многообразие форм в естественных языках, чтобы разработать их функциональное использование, ответить на вопрос, почему человек использует одну форму, а не другую, а далее формализовать этот процесс.

Второй вопрос - это контроль над процессом порождения. Что определяет выбор говорящего в данной языковой ситуации? Как человек организовывает и представляет промежуточные результаты? Какими знаниями о зависимостях между вариантами выбора должна обладать система? Как представлены эти зависимости и как они могут влиять на алгоритмы управления? Ответы на поставленные вопросы будут рассмотрены в этой статье.

Контроль над постепенной обработкой сообщения. Среди порождающих систем, которые были специально построены для работы в основных системах, преобладающий подход контроля состоит в обработке сообщений как определенного вида программ. Эти "сообщения" не просто выражения, чьи контекст и форма изоморфны по отношению к конечному тексту. “Сообщения” могут быть закодированы на компьютерном языке. Их нельзя просто перевести. Конечно, при самой простой обработке порождения, перевода было бы достаточно (как почти во всех существующих ЭС), но в обработке, которая сосредоточена на порождении текстов на естественном языке, отношения и содержание в сообщении лучше всего просматриваются в виде команд для достижения определенного эффекта лингвистическими средствами. Оценка происходит при постепенной обработке от внешних команд к внутренним. Эта методика контроля естественна для разработчиков систем, так как она имитирует стиль языков программирования, которые они используют.

Наиболее общие сообщения сегодня не создаются планировщиком, а являются просто структурами данных, которые извлекаются из основной программы и которым генератор дает особую интерпретацию. Подобная практика распространена в программах, которым необходимо объяснять свои рассуждения, заключенные в доказательстве дедуктивным методом исчисления предиката. Ниже приводится такого рода доказательство.

 На входе

Line 1: premis

 Exists(x) [barber(x) and

 Forall(y)..shaves(x,y) iff not.shaves(y,y)l

Line 2: existential instantiation (1)

 barber(g)and Forall(y)..shaves(g,y) iff not.shaves(y,y)

Line 3: conjunction reduction (2)

 Forall (y)..shaves(g,y) iff not.shaves(y,y)

Line 4: universal instantiation (3)

 shaves(g,g) iff not.shaves(g,g)

Line 5: tautology (4)

 shaves(g,g) and not.shaves(g,g)

Line 6: conditionalization (5,1)

 (Exists(x) [barber(x) and

 Forall(y)..shaves (x,y) iff not.shaves(y,y)]

 implies (shave(g,g) and not.shaves(g,g))

Line 7: reductio-ad-absurdum (6)

 not(Exists(x) barber(x) and

 Forall(y)..shaves (x,y)

 iff not.shaves(y,y))

На выходе

Assume that there is some barber who shaves everyone who doesn’t shave himself (and no one else). Call him Giuseppe. Now, anyone who doesn’t shave himself would be shaved by Giuseppe. This would include Guiseppe himself. That is, he would shave himself, if and only if he did not shave himself, which is a conradiction. Therefore it is false, there is no such barber.

Модель дает объяснение действиям автора доказательства в выборе, какое правило применять, например, что цель правой части условия в первой строке наложить ограничение на переменную Y ("... Кто не бреет себя "). Это дает право воспринимать доказательство особым образом. Эти действия, однако, нигде в доказательстве (которое было единственным входом в программу) не появляется. Они только предполагаются и, таким образом, имеют силу только для нескольких примеров доказательств, произведенных естественным дедуктивным методом.

Недостаток информации в сообщениях основной программы - постоянная проблема в работе с порождением текстов. Специалисты по вычислительной лингвистике вынуждены вчитываться в структуры данных основных программ, потому что последние уже не включают те виды риторических команд, которые необходимы генератору, если следовать синтаксическим конструкциям языка, которые использует человек. Без “дополнительной” информации связность произносимого - особенно для длинных текстов - будет зависеть от того, насколько непротиворечиво и полно авторы основных программ представили информацию: каждый раз, когда генератор встречает к-л символ, ему ничего не остается как обрабатывать его как "посылку" или как условие одним и тем же способом, если он встречает их в одинаковом контексте. Если поддерживается непротиворечивость, проектировщик может восполнять неточности, усовершенствуя структуры данных, как только они оказываются внутри лингвистического компонента.

Средства, направленные на достижение беглости и преднамеренной детализации формы, объясняют использование фразовых словарей и промежуточного лингвистического представления. Простой пример показывает, почему это необходимо. Рассмотрим логическую формулу, которую программа обычно использовала бы внутренне. В этом примере обработка проводится тем же методом, что описан выше. Пример представляет из себя наиболее общий вид сообщения: выражение прямо из модели основной программы (система доказательства естественным дедуктивным методом), которому теперь дается особая интерпретация, так как это выражение служит для анализа текста.

(exists x

 (and barber(x)

 (forall y

 (if-and-only-if shaves(x,y)

 (not shaves(y, y) )))))

В этой формуле генератор одновременно сопоставляется с выбором реализации. Должно ли навешивание кванторов выражаться буквально ("Существует такой X, что ..."), или должно быть свернутым внутри основной части как определяющая информация относительно реализации переменных ("...some barber”)? Должно ли условие if-and-only-if реализовываться буквально как конъюнкция подчинения или может быть интерпретировано как ограничение диапазона переменной? Утверждение типа barber(x), по-видимому, всегда должно декодироваться и преобразовываться в детальное описание переменной. Остальное реализуется независимым образом, однако, после тщательного обдумывания.

Объекты, которые заполняют "мозг" основной программы, в данном случае - логические связки, предикаты, и переменные, полностью связаны со словами и грамматическими конструкциями, которые подлежат обработке "специальными процедурами/ процедурами знаний" поддерживаемыми внутри генератора. Эти процедуры - эквивалент словаря в понимающей системе. Специалисты строят фразу для понимания, используя лексическую информацию, связанную непосредственно с индивидуальными логическими объектами. Каждый объект обычно ассоциируется с к-л лексическими единицами: константа может иметь имя; предикат может иметь прилагательное или глагол. Специалист помещает их во фразовый контекст, который будет дополнен рекурсивной прикладной программой других специалистов, например, двуместный предикат "shaves(x,y)" становится шаблоном предложения "x shaves y."

Таким образом, лингвистические шаблоны обеспечивают упорядоченную реализацию параметров, что поддерживает эффективное функционирование с наименьшим количеством блокирований, ускоряя процесс порождения в целом, избегая необходимость "резервировать" преждевременные решения, которые могут оказаться несовместимыми с грамматическим контекстом, определенным более высоким шаблоном.

Лексический Выбор. Некоторые подходы к машинному пониманию основываются на небольшом наборе базисных элементов (qv) и, формулируют знания программы в виде набора выражений к базисным элементам, что упрощает работу программы: становится легче выводить умозаключения, потому что при помощи базисных элементов они распределяются в естественные группы. Однако, сведение диапазона человеческих действий к определенному набору, например, лишь к 13 концептуальным базисным элементам, означает, что специфика значений распределяется в выражениях и извлекается оттуда каждый раз, если во время порождения необходимо использовать глаголы со специфическим значением. Голдман первый провел исследования по использованию сетей распознавания. Он показал, как производится выбор слова, в отрыве от основных базисных элементов. Например, из базисного элемента действия "глотать" можно получить глаголы "пить", "есть", "вдыхать", "дышать", "курить", или *"проглотить"*, как бы проверяя при этом, был ли проглоченный объект жидкостью или дымом.

Проект сети распознавания заставляет исследователя порождения выходить за рамки основных различий типов объектов и включать контекстные факторы, напр., эмоциональные рассуждения говорящего. Ниже - выборка из работы Хови, цель которой состояла в том, чтобы сместить текст, чтобы подчеркнуть желаемую точку зрения (в данном случае сообщить в февральских первичных выборах так, чтобы результаты понравились Картеру, даже если он проиграл.

Kennedy only got a small number of delegates in the elections on 20 February. Cater just lost by a small number of votes. He has several delegates more than Kennedy in total.

Фразовые словари. Какое слово ассоциируется с простыми понятиями, типа "парикмахер" или "брить", является очевидным; однако, для объектов в комплексных основных программах, лексический выбор может оказаться более проблематичным. Помощь в этой ситуации может оказать использование фразового словаря. Это понятие было введено в 1975 Бекером и с тех пор стало важным инструментом систем порождения. С лингвистической точки зрения, "фразовый" словарь - ­концептуальное расширение стандартного словаря, включающее все непроанализированные фразы, - на той же самой семантической основе, что и словарь отдельных слов. Это обеспечивает фиксацию незаконсервированных идиом и различных речевых способов, которые люди используют каждый день. Так как люди используют эти " фиксированные фразы " как нерасчленимое целое, программы должны научиться делать то же самое. Пример ниже - из работы Кукича.

Wall Street securities markets meandered upward through most of the motning, before being pushed downhill late in the day yesterday. The stock market closed out the day with a small loss and turned in mixed showing in moderate trading.

Это информационное объявление было вычислено непосредственно из анализа данных по поведению рынка в течение дня. Качественные моменты в сообщении были соединены непосредственно со стереотипными фразами подобного рода объявлений: "a small loss", "a mixed showing", "in moderate trading". Объекты, действия и указатели времени были отображены непосредственно в соответствующих цепочках слов: "Wall Street securities markets", "meandered upward ", "be pushed downhill", "late in the day". Композиционный шаблон состоит из предложений, сформированных на основе S-V-Advp фразы: (рынок) (действие) (указатель времени).

Обработка Грамматики

В изучении порождения выбор формализации представления грамматики языка всегда связывался с выбором протокола контроля. Известны три основных подхода к решению этого вопроса:

1. грамматика как независимый корпус предложений и фильтр к ним (например, *объединенная функциональная грамматика*);

2. использование грамматики с целью выявления всех возможных поверхностных структур, доступных для языка; затем проведение выбора и реализации среди данных поверхностных структур (смысловые подходы);

3. грамматика как структура пересеченного графа, который контролирует весь процесс, как только создается план текста (план выражения) (грамматика расширенных сетей переходов, а также систематическая грамматика).

В этой статье не оказывается предпочтение ни одному из трех подходов. Однако каждый из них будет рассмотрен в соответствие с поставленной задачей, которая мотивирует использование этих подходов.

Объединенная Функциональная Грамматика (ОФГ) в порождении.

Объединенная Функциональная Грамматика была разработана Кейем, является “реверсивной” грамматикой, т. е. может использоваться как при порождении, так и при понимании речи.

Термин "функциональный", по мнению разработчиков, говорит о том, что следует оттолкнуться от описания структуры лингвистических форм, чтобы обратиться к причинам, почему используется язык. В отличие от систематических грамматик, *функциональные элементы* в ОФГ представляют к настоящему времени лишь минимальное расширение стандартного категориального лингвистического словаря, используемого традиционно, чтобы описать синтаксическую форму (например, "clause", "noun phrase", "adjective"), и имеют много общего с "лексико-функциональной грамматикой", (стоящей в той же парадигме грамматик). Классическое *функциональное* значение, типа различие между "уже имеющейся" и "новой" информацией в предложении, подобно различию между "темой" и "ремой”, еще не включено в ОФГ. ОФГ использует “telegram” грамматику, разработанную Аппельтом, понимающий компонент, написанный Босси.

 Первый пример (из Аппельта) описывает одну из составляющих ролей, которые сопровождают фразовую категорию, именную фразу.

ОФГ используют, чтобы изложить в деталях минимальные, концептуально полученные функциональные описания, например, что главным словом к-л именной фразы должно быть слово "отвертка". Недавняя работа Паттена использует систематическую грамматику в очень схожим образом. Операции такого типа на семантическом уровне, выполняемые в других подходах путем планирования уровня, специалисты определяют как набор особенностей *вывода* внутри систематической грамматики, эквивалент начального функционального описания, которое управляет ОФГ. Обратное и прямое формирование цепочки перемещается через систематическую грамматику, затем определяет, какие дополнительные лингвистические особенности должны быть добавлены к грамматической спецификации текста.

ОФГ используются в процессе последовательных объединений, ограниченных правилами, которые следят за тем, как два описания могут быть объединены. Ключевая идея состоит в том, что планировщик первоначально создает минимальное описание фразы, что можно делать и стандартным способом. Чтобы излагать в деталях описание к пунктам, где это было бы грамматически верно, оно затем объединяется с грамматикой: описание фразы и спецификация грамматики успешно объединены. Конкретизация понятий прежде не определенных особенностей описания константами, снабженными грамматикой, вызывает эффект ряби во всей системе: решения, которые зависят от только что конкретизированных особенностей, провоцируют дальнейшее циклическое объединение, пока не будет сформулировано грамматически полное описание высказывания. Кроме того, элементы в описании планировщика побуждают к отбору среди дизъюнктивных спецификаций в грамматике. Например, определение глагола приводит к выбору грамматической подклассификации.

Полное описание составляет дерево подописаний (составляющих) как определено "стандартом" (образцом), который предписывает последовательный порядок на каждом уровне. Фактически текст создается при просмотре этого дерева и чтении слов с лексическими особенностями каждой составляющей. Ограничения накладываются в процессе объединения: только совместимые частичные описания присутствуют в конечном результате. Это имеет большое значение, так как планировщику не нужно разбираться с грамматическими ограничениями и зависимостями, что, с другой стороны ограничивает его потенциал: он не может пользоваться знаниями по грамматическим ограничениям, даже когда ему это понадобится.

С точки зрения разработки грамматики, ОФГ является вполне удовлетворительной, так как данный подход позволяет компактно формулировать языковые факты, то есть необязательно расшифровывать взаимосвязь между предложениями, так как это происходит автоматически во время объединения.

Прямой Контроль Грамматики при Понимании: Систематическая Грамматика и Грамматика Расширенных Сетей Переходов (РСП). Расширенная сеть переходов используется в порождении почти с момента своего определения. РСП использовали сначала Симмонс и Слокум в 1970, чью систему затем использовал Голдман. РСП также применял Шапиро, чей генератор, в этой группе, является наиболее продуманным. Все системы имеют схожую структуру. Они просматривают структуру данных, которую поддерживает основная программа. Сети поддерживают формат сверху-вниз, как обычно у всех РСП-парсеров (синтаксических анализаторов). Для ранних РСП подобная структура являлась семантической сетью, основанной на теории фреймов с глаголом в центральной части (еще одна "функциональная" лингвистическая система). Специальный узел в сети, "вектор модальности", определяет информацию на корневом уровне, например, время и вид; является предложение активным или пассивным. Первичная функция РСП в ранних системах состояла в линейном упорядочении сетевой структуры, которая была главным образом уже закодирована в лингвистическом словаре.

РСП, по существу, представляет из себя процедурное кодирование порождающей грамматики. Регистры, которые дают сетям "расширенное" влияние, используются как представление грамматических отношений с глубинной структурой, и пути в сетях кодируют все составные поверхностные альтернативные последовательности. Ограничения распространяются по дереву сверху-вниз (то есть к рекурсивным подсетям РСП) через значения в обозначенных регистров, приводя в действие подсети при контекстном управлении. Проект РСП Шапиро особенно впечатляет, поскольку его структура управления данных занимает весь вычислительный режим основной программы.

Дальнейший аспект проекта РСП - тот факт, что средства создания слов текста являются выполнением побочного эффекта по прохождению ребра графа, что приводит генератор к действию почти в тот момент, когда ситуация воспринимается. Особенно впечатляет то, что оценивает, что РСП Шапиро никогда не пользуется резервированием. Это - совершенно необычное поведение для РСП, так как порождение является в сущности процессом планирования.

 Наиболее значительной проблемой для проектов РСП - ­трудность выделения понимания из действия. Генераторы, основанные на систематической грамматике, имеют дело с этой проблемой, непосредственно представляя срединную репрезентацию в форме набора характерных признаков, что позволяет спецификации текста постепенно накапливаться, предоставляя ограничениям возможность распространяться и влиять на более поздние решения.

Две важных системы порождения были основаны на систематической грамматике: PROTEUS Дэйви(обсуждали ранее) и NIGEL Манна и Маттхиссена. NIGEL - самая большая систематическая грамматика в мире и, очень вероятно, одна из самых больших машинных грамматик любого сорта.

РАСПОЗНАВАНИЕ РЕЧИ.

 По мере развития компьютерных систем становится все более очевидным, что использование этих систем намного расширится, если станет возможным использование человеческой речи при работе непосредственно с компьютером, и в частности станет возможным управление машиной обычным голосом в реальном времени, а также ввод и вывод информации в виде обычной человеческой речи.

 Существующие технологии распознавания речи не имеют пока достаточных возможностей для их широкого использования, но на данном этапе исследований проводится интенсивный поиск возможностей употребления коротких многозначных слов (процедур) для облегчения понимания. Распознавание речи в настоящее время нашло реальное применение в жизни, пожалуй, только в тех случаях, когда используемый словарь сокращен до 10 знаков, например при обработке номеров кредитных карт и прочих кодов доступа в базирующихся на компьютерах системах, обрабатывающих передаваемые по телефону данные. Так что насущная задача - распознавание по крайней мере 20 тысяч слов естественного языка - остается пока недостижимой. Эти возможности пока недоступны для широкого коммерческого использования. Однако ряд компаний своими силами пытается использовать уже существующие в данной области науки знания.

 Для успешного распознавания речи следует решить следующие задачи:

обработку словаря (фонемный состав),

обработку синтаксиса,

сокращение речи (включая возможное использование жестких сценариев),

выбор диктора (включая возраст, пол, родной язык и диалект),

тренировку дикторов,

выбор особенного вида микрофона (принимая во внимание направленность и местоположение микрофона),

условия работы системы и получения результата с указанием ошибок.

 Существующие сегодня системы распознавания речи основываются на сборе всей доступной (порой даже избыточной) информации, необходимой для распознавания слов. Исследователи считают, что таким образом задача распознавания образца речи, основанная на качестве сигнала, подверженного изменениям, будет достаточной для распознавани, но тем неменее в настоящее время даже при распознавании небольших сообщений нормальной речи, пока невозможно после получения разнообразных реальных сигналов осуществить прямую трансформацию в лингвистические символы, что является желаемым результатом.

 Вместо этого проводится процесс, первым шагом которого является первоначальное трансформирование вводимой информации для сокращения обрабатываемого объема так, чтобы ее можно было бы подвергнуть компьютерному анализу. Примером является “техника сопоставления отрезков”, позволяющая сократить вводимую информацию с 50'000 до 800 битов в секунду. Следующим этапом является спектральное представление речи, получившееся путем преобразования Фурье. Результат преобразования Фурье позволяет не только сжать информацию, но и дает возможность сконцентрироваться на важных аспектах речи, которые интенсивно изучались в сфере экспериментальной фонетики. Пример такого представления см на рис. Спектральное представление достигнуто путем использования широко-частотного анализа записи.

 Хотя спектральное представление речи очень полезно, необходимо помнить, что изучаемый сигнал весьма разнообразен. Разнообразие возникает по многим причинам, включая:

различия человеческих голосов;

уровень речи говорящего;

вариации в произношении;

нормальное варьирование движения артикуляторов (языка, губ, челюсти, нёба).

Для устранения негативного эффекта влияния варьирования голосового тракта на процесс распознавания речи было использовано множество методов. Первым делом рассматривалась характеристика пространства траектории артикуляторных органов, включая гласные, используемые говорящим. Наиболее удачные формы трансформации, использованной для сокращения различий, были впервые представлены Сакоя & Чибо и назывались динамичными искажениями (dynamic time warping). Техника динамичного искажения используется для временного вытягивания и сокращения расстояния между искаженным спектральным представлением и шаблоном для говорящего. Использование данной техники дало улучшении точного распознавания (~20-30%). Метод динамичного искажения используют практически все коммерчески доступные системы распознавания, показывающие высокую точность сообщения при использовании. Техника динамичного искажения представлена на рис.2. Вначале сигнал преобразовывается в спектральное представление, где определяется немногочисленный, но высокоинформативный набор параметров. Затем определяются конечные выходные параметры для варьирования голоса(следует отметить, что данная задача не является тривиальной) и производится нормализация для составления шкалы параметров, а также для определения ситуационного уровня речи. Вышеописанные измененные параметры используются затем для создания шаблона. Шаблон включается в словарь, который характеризует произнесение звуков при передаче информации говорящим, использующим эту систему. Далее в процессе распознавания новых речевых образцов (уже подвергшихся нормализации и получивших свои параметры), эти образцы сравниваются с шаблонами, уже имеющимися в словаре, используя динамичное искажение и похожие метрические измерения. В настоящее время этот метод изучается и дополняется.

 Очевидно, что спектральное представление речи позволяет характеризовать особенности голосового тракта человека и способ использования его говорящим. Самый обычный способ моделирования специфических эффектов "модель-источник" - использование фильтров. Речевой аппарат моделируется с использованием источников, вызывающих резонанс, ведущий к пиковым точкам интенсивности звука в соседстве с отдельными частотами, называемыми формантами. При произнесении звуков вибрация голосовых связок является источником возбуждения, и эти короткие импульсы вызывают резонанс между голосовыми связками и губами. Так как язык, челюсть, губы, зубы и альвеолярный аппарат двигаются, размер и место этих резонансов меняются, давая возможность воспроизведения особых параметров звуков.

 Возможно построить очень точную модель, также прямо смоделировать движения артикуляторов физиологически реальным путем. Использование этих моделей привели к пониманию пути, в котором происходит речевой сигнал. Но так как наблюдение над артикуляторами затруднено, остаются недостатки. Хотя природа вокального тракта очень сильно влияет на выходной сигнал речи, это не единственное ограничение, которое необходимо принимать во внимание, так как контроль над мускулами звукового тракта обусловлен сигналами моторного кортэкса мозга. Возможно все аспекты влияния акустической структуры контролируют сигналы и форму звукового выхода речи (хотя это не может быть доказано с систематической точки зрения).

Аспекты влияния акустической структуры включает в себя:

природу сегментов индивидуального звука (гласные/согласные),

структуру слога,

структуру морфем (приставки, корни, суффиксы),

лексикон,

уровень синтаксиса фраз и предложений и

долгосрочные ограничения речи (long-term discourse constraints) .

 Ниже рассматривается влияние ограничений и способ их воздействия производство сигнала речи. Необходимо также принять во внимание тот факт, что человеческий аппарат восприятия также должен быть смоделирован, он сам по себе накладывает на процесс восприятия дополнительные ограничения. Недавно процесс восприятия был изучен с помощью метода сигнального подавления барабанных перепонок через возбуждение нервных клеток, которые образовывают примерно 30 тысяч нервных окончаний слухового нерва. Но изучение нервных окончаний способно только прояснить формирование простых синтетических гласных. Перед исследователями встало новое главное направление в области изучения воспроизводства речи, связанное с интеграцией всей физиологии восприятия человека. В настоящий момент появляются некоторые модели явлений, происходящих в ухе, и не без оснований можно ожидать дальнейшего улучшения понимания процесса распознавания речи из-за более полного понимания характеристик этого влияния.

 Что касается уровня артикуляторного контроля, первым уровнем является индивидуальный фонетический сегмент, иначе говоря, - фонема. Во многих естественных языках их примерно 40. Но их набор существенно различатется. Поэтому, например, английские гласные могут быть носовыми, даже ненамеренно, в то время как во французском носализация гласных является фонетическим контрастом, и поэтому влияют на значение произносимого. Во французском языке носовая коартикуляция доминирует в гласных и существенно влияет на восприятие фонем и следовательно на главный смысл значения. Хотя все говорящие имеют одинаковый голосовой аппарат, использование его разное. Так например, использование кончика языка или прищелкивание, как в некоторых африканских языках. Ясно, что природа артикуляционных движений имеет сильное влияние на метод воспроизведения речи. Эти ограничения всегда активно используются в практических системах.

 На следующем уровне лингвистической структуры фонетические сегменты сгруппированы в согласные/гласные, а следовательно и в слоги. Далее, в зависимости от роли фонетического сегмента внутри этих слогов их реализация может быть сильно изменена. Так например, начальный согласный в слоге может быть реализован как абсолютно отличный от конечной позиции. Согласные очень крепко связываются между собой, что опять же влияет на последующие ограничения. Например, в английском если начальная группа согласных состоит из трех фонем, первая фонема должна быть /s/, следующей фонемой должен быть непроизносимый согласный, третьей или /r/ или /l/, как например, в слове /scrape/ или /split/. Говорящие на родном языке избегают этих ограничений или могут активно их использовать во время процесса восприятия. Из выше приведенных примеров очевидно, что хотя и существуют сильные ограничения, влияющие на слушателя, но их сила не является решающей во время произнесения речи. То есть любое моделирование процесса восприятия может быть активным и может оказать большую помощь в понимании главного смысла.

 Другой пример, показывающий необходимость применения сфокусированного поиска, может быть представлен в восприятии конечного согласного. Среди многих ключевых слов для распознавания конечного согласного существует спектральная природа шума, воспроизводимого при освобождении конечной перемычки и перехода резонанса второй форманты в гласный, следующий за этой перемычкой. Многие исследователи изучали эти влияния, и результаты их исследований показали, что ограничивающее влияние обоих вышеописанных характеристик на восприятие варьируется природой следующего гласного, и следовательно, мощная стратегия распознавания должна иметь некоторые знания о твердой позиции гласного перед конечным согласным перед тем, как будет сделано само распознавание конечного согласного. Конечные согласные дают яркий пример весьма интересного комплекса фонетики, используемого для лингвистической окраски. Например, при рассмотрении слов rapid и rabid обнаруживается 16 фонетический различий.

 Кроме сегментного и слогового уровней существуют ограниченные влияния из-за структуры морфем, которые являются минимальными синтаксическими единицами языка. Они включают в себя приставки, корни, суффиксы. Можно себе представить, что это синтаксис на слоговом и на морфемном уровнях, также как и нормально распознанный синтаксис, характеризующийся способом, в котором английские слова объединяются во фразы и предложения. Возможно представить данные ограничения как последствия рассмотрения грамматики вне контекста. В этом виде ограничений много “шумных” вариаций сегментов речи, которые так же относятся и к иерархическим синтаксическим ограничениям.

 Дополнительные ограничения на природе входа новой лексики в язык могут являться уровнем слова. Многие исследования обнаружили, что характеристика слов при введении разбиения на 5 жестких классов фонетических сегментов может быть сокращена до минимума, часто имея единственное в своем роде распознавание. Далее слишком усиливается эффект порядка двух букв и фонетических сегментов с тех пор как в изучении английских и французских словарей было обнаружено, что более 90% слов имели единственное значение и только 0,5% имели 2 и больше альтернатив. На фонемном уровне было обнаружено, что все слова в английском словаре из 20 тысяч слов имели одно значение из-за беспорядочных фонемных пар. Этот пример помогает показать, что все еще существует ограничивающее влияние на лексическом уровне, которое еще не определено в современных системах распознавания речи. Естественно, что исследования в этой области продолжаются.

 Кроме уровня слов синтаксис имеет дополнительное ограничительное влияние. Его влияние на последовательный порядок слов часто характеризуется в системах фактором, который в свою очередь характеризует количество возможных слов, которые могут следовать за предыдущим словом в процессе произнесения. Синтаксис также имеет ограничительные влияния на просодические элементы, такие как ударение, например в случае, когда ударение слов в incline и survey варьируется в зависимости от части речи. Возможно для того, чтобы охарактеризовать ударение в слове, нужно принять во внимание не только индивидуальное слово, но вышеприведенные дополнительные ограничения синтаксиса.

 Далее, кроме синтаксического уровня ограничения доминируют над семантикой, прагматикой и речью, что плохо осознается людьми, однако имеет очень важное значение для процесса распознавания.

 Несмотря на сложность описания характеристик источников различных ограничений, немаловажную роль играют современные системы влияния, которые представлены всеми возможными вариантами произнесения звуков. Например, система HARPI университета Сarnegie-Mellon University является системой, в которой звуковоспроизведение описывается как путь через комплексную сеть. В этом способе ограничения структуры слога, слова и синтаксиса связаны одной структурой. Структура контроля, используемая для поиска, является адаптацией динамичной программной техники. Более сильный подход был предложен моделями использования цепей Маркова. Эти модели использовались как единая структура, где возможности могут быть точно изучены экспериментальным путем. Закодированные представления спектральной трансформации воспроизводства речи используются для нахождения самого правильного пути через сеть, и недавно были получены очень хорошие результаты. Очень важно подчеркнуть использование такого формально- структурного подхода, который способствует автоматичному определению классов символов через структурирование и параметризацию.

 При другом подходе базы данных и связанные с ними процессы обработки используются структурой контроля. Этот подход был изучен системой HEARSAJ 2, которая была разработана в институте Сarnegie-Mellon University, и системой HWIM (hear what I mean). В этих системах комплексная структура данных, которая содержит всю информацию о воспроизведении звуков, изучается с точки зрения конкретных ограничений. Но как выше указано, каждое из этих ограничений имеет особую внутреннюю модель, и полный анализ не может быть произведен. Для проведения анализа в целом структура данных должна иметь взаимодействие между разными процессами, а также средства для интеграции. Несмотря на то, что структура включает в себя несколько весьма различных источников знаний и ее вклад в понимание речи очень общий, она также имеет большое количество степеней свободы, которые могут быть использованы для тщательного системного воспроизведения. В отличие от этого, техника, основанная на цепях Маркова, имеет математическую поддержку. Чтобы иметь возможность сфокусированного исследования ограничений взаимодействия и интеграции в контексте, необходимо применять обе системы. Те системы, которые описывают ограничение взаимодействия, сфокусированы во многом на воспроизведении знаний, и они относительно слабо контролируемы, а системам с математической поддержкой, которые в свою очередь имеют великолепную технику для установления параметров и оптимизации изучения, не достает использования комплексной структуры данных, необходимых для характеристики ограничений высокого уровня, таких как синтаксис. Оба направления в настоящий момент находятся в процессе развития.

 В заключение следует сделать акцент на влияние производственной технологии на эти системы. Технология интеграции не является большой проблемой для систем распознавания речи, наоборот, это является архитектурой этих систем, включая способ представления ограничений. Необходимо провести грандиозные эксперименты и найти новые способы, которые необходимы для ограничительного влияния взаимодействия.

 Во многих способах распознавание речи имеет типичный пример стремительно развивающегося класса высоко интегрированных комплексных систем, которые должны использовать лучшую компьютерную технику и самые последние достижения современного математического обеспечения.

Морфологический анализ научного текста несловарного типа

В современных системах МА существуют два основных принципа выведения морфологических признаков слова с помощью анализа его структуры:

Представление грамматической информации в словаре основ и в словаре флексий (с включением формальной процедуры деления слова на основу и окончание и последующее сравнение со словарями).

Извлечение грамматической информации из слова путем его графемного анализа.

В описываемой системе, представляющей собой подсистему автоматического анализа научно-реферативного текста, выбирается второй принцип. Используется он на первом этапе анализа, допускающем приписывание единицам текста дизъюнктивных кодов, репрезентирующих грамматические омонимы. Обусловлено это тем, что любой морфологический анализ (МА), проводимый в пределах слова, не может обеспечить стопроцентного однозначного определения его морфологических характеристик, поскольку проявление последних в действительности происходит в синтагматике, на уровне связей слов в предложении. Поэтому в системе МА предусмотрен второй этап анализа, который корректирует результаты первого с помощью анализа грамматического контекста единиц с дизъюнктивными кодами. Кроме того, и на первом, и на втором этапах МА в отдельных случаях возникает необходимость обращения к информации о значении слова, например для снятия грамматической омонимии существительных. Предлагается задание списка лексем или отдельных лексем в виде цепочек начальных буквосочетаний слова, общих для всех словоформ. Возможность задавать семантическую информацию таким образом вытекает из ограниченности лексического состава текстов анализируемого подъязыка (в тексте длиной 108 тысяч словоупотреблений различных лексем знаменательных лексем насчитывается всего 6653). Доля использования семантической информации при флективном анализе – 11,5 %, при контекстном анализе – 9,1 %.

Общие задачи МА можно определить следующим образом:

Однозначное идентифицирование единиц текста в терминах лексико-грамматических классов.

Определение внутри класса словоизменительных характеристик словоформ (грамматический подкласс).

Сведение словоизменительных парадигм, приведение словоформ одной лексемы к канонической форме.

Работе алгоритмов собственно морфологического анализа предшествует процедура предредактирования вводимых в машину текстов, ориентированная не только на нужды МА, но и на последующие задачи всей системы автоматического анализа текста. Кроме разбиения текста на машинные слова (слова в форме, удобной для распознования их машиной): пробелами отделяются все знаки препинания, кроме тех случаев, когда они выступают не в своей основной синтаксической функции, текст должен быть также размечен: заглавия реферата и абзацев внутри него. В данной системе приходится еще редактировать текст, приводя в соответствие множество символов естественного языка и представления их в машине. Вместо символов, отсутствующих в устройстве ввода/вывода, ставятся особые пометы, например заглавные буквы, если они не относятся к начальному слова предложения, отмечаются звездочкой и знаком >, если в слове больше одной заглавной, то впереди ставится цефра, указывающая на их количество: *7>фортран (=ФОРТРАН)*.

Записанный в такой форме текст поступает на вход системы МА. Данная система состоит из пяти подсистемы. В задачи подсистемы 1 входит:

А) выделение тестовых единиц, требующих собственного МА;

Б) анализ слов, не имеющих морфологического статуса, типа формул, сокращенных словосочетаний, слов из букв латинского алфавита; Морфологическим статусом обладают все слова, имеющие в совем составе только русские прописные буквы или ограниченное кол-во знаков препинания, как-то дефис, косая черта и т.п., а также слова, ничинающиеся цепочкой цифр или латинских букв, за которыми следует дефис, и последефисная часть слова состоит из русских прописных букв: 15-й.

В) выявление ошибок, допущенных при перфорации текста.

Подсистема 2 предназначена для обработки текстовых единиц, получивших в процессе предредактирования метку заглавной буквы. Используется открытый список наиболее частотных аббревиатур (ЭВМ) и некоторые словоупотребления, сокращения

Подсистема 3 определяет коды грамматических классов для слов на основе анализа их графемной структуры.

Подсистема 4 определяет коды грамматических подклассов в пределах класса с помощью графематического анализа. Алгоритмы определения подклассов разные для местоимений, наречий, кратких причастий и прилагательных, существительных, слов адъективного типа. Например, род существительного определяется с вероятностью 99,98 % по графемному составу основы. В работе алгоритма определения подкласса существительного каждой словоформе присваивается двухсимвольный код, первая позиция – род, вторая число и падеж, всего выделяются 42 однозначных подклассов и 34 дизъюнктивных (типа компонент – м.р./вин. ед. и ж.р./ род. мн.).

В функции подсистемы 5 входит снятие грамматической омонимии классав и подклассов, полученной в результате работы алгоритмов 2,3,4, на основе контекстного анализа.

В автономном режиме работает только контекстный анализ. Его работа начинается после того, как все слова текста обработаны предшествующими алгоритмами и на вход поступает текст, в котором каждое машинное слово, отличное от знака препинания, снабжено двухбуквенным кодом, первая из которых – код класса, вторая – код подкласса. Коды грамматических омонимов содержат в себе информацию о возможных грамматических значениях данной словоформы, например,

Р – существительное/предлог (*путем*),

D – существительное/глагол (*начала, суть*),

Ы- союз/частица/наречие (только) и т.д.)

Основная часть алгоритмов несловарного морфологического анализа сосредоточена в третьей подсистеме, которая должна единицы текста в терминах грамматических классов на основе списка квазифлексий.. Ограниченность лексической системы используемого подъязыка и типа текстов и делает возможным применения принципа определения грамматической информации по квазифлексиям в качестве основного инструмента МИ и отказ вообще от словаря основ как главной базы МА. На вход данного алгоритма после работы подсистем 1,2 поступает 93,54 % текстовых единиц.

Автоматическое построение списка квазифлексий для морфологической идентификации словоформ в тексте предполагает:

Ручное построение обучающей выборки на фрагменте введенного в ЭВМ текста. Для ручной частеречной кодировки принята детализированная система классов слов:

существительное--И

полное прилагательное --Я

полное причаcтие--Е

глагол--Г

наречие--Н

краткое прилагательное --Ю

краткое причаcтие--У

нераспознанные по алгоритму слова, и т.п.

Омонимичные словоформы типа *для ее обработки и стобы ее обработать* разграничиваются.

Автоматическое построение инверсионного словаря и автоматический вывод списка квазифлексий

Ручную проверку и уточнение результатов автоматической морфологической идентификации словоформ в новом тексте

Автоматическое пополнение инверсионного словаря словоформ обучающей выборки автоматическую перестройку списка квазифлексий.

При контекстном анализе КА, разработанных для снятия грамматической омонимии отдельных слов по окружению, наибольшей диагностицирующей силой обладают грамматические контексты, представленные предлогами, контекстами, краткими причастиями и прилагательными, существительными, подчинительными союзами и местоимениями и наречиями-союзных слов, а также знаки препинания. Все названные элементы текста используются во всех без исключения алгоритмах КА, в качестве опорных точек ОТ. При этом дизъюнкция класса/подкласса может сниматься по пересечению содержащейся в нем информации с грамматической информацией ОТ, образующей левую или правую, а также обе границы (снятие омонимии падежа существительного по предлогу слева, снятие омонимии классов Z (полное прилагательное/наречие); Y (краткое прилагательное/наречие) по расположенному справа глаголу; анализ дизъюнктивных кодов словоформ, принадлежащих к адъективным классам).

Алгоритм включает разные ситуации, как-то именная ситуация(SIT NOUN), глагольная(SIT VERB), предложная (SIT PREP), ситуацию аббревиатуры (SIT U) и прочие. Опорные точки могут образовывать ситуацию в сочетании с нектороми грамматическими классами, например ситуация POINT определяется тире + частица *ЭТО,* запятая + наречие *когда, где.* Анализ производится слева направа, начиная с первого элемента текста, и состоит из двух частей: входа, формирующего ситуацию и собственно контектного анализа, при этом ищутся опорные точки, формируется ситуация и по правилам КА снимаются дизъюнктивные классы.

Анализ результатов машинных экспериментов работы описаннойй системы МА на рефератах по кибернетике и системам связи показал высокую степень эффективности сочетания флективного и контекстного анализа при автоматическом МА, ориентированном на подъязык научно-реферативных текстов широкой тематики. Неразграниченные омонимы грамматических классов составляют всего 3,93 % текста. Количество словоформ с неснятой омонимией подклассов равно 10,34 %. Это объясняется, прежде всего, необходимостью привлечения анализа целого предложения (краткое прилагательное в роле подлежащего) и анализа контекстных связей, выходящих за пределы предложения (анализ эллиптических конструкций, снятие омонимии местоименных классов). Ниже приводится пример машинной кодировки, полученной после флективного и контекстных анализов. При идентификации грамматического стутуса каждого класса и подкласса слов в алгоритме МА используется различное количество правил и разные типы правил. В основу системы правил МА положена языковая и текстовая информация разного типа, в том числе:

графемная структура словоформ.

Позиционные закономерности употребления словоформ в тексте.

Дистрибуция словоформ и их соположение в тексте.

Дистрибуция словоформ, учитывающая их грамматические связи.

Лексическая информация.

АВТОКОРРЕЛЯЦИОННАЯ(ЯЖ) СТРУКТУРА(КИ) ГЕНЕРАТОРА(ЙР) СЛУЧАЙНЫХ(КЕ) ЧИСЕЛ(ЛЕ) 1>ТАУСВОРТА(ЙР).

ОСНОВЫВАЯСЬ(Д) НА(ПП) ЗНАНИИ(ЛП) АВТОКОРРЕЛЯЦИОННЫХ(ЯЕ) ФУНКЦИЙ(КЕ) ДЛЯ(ПР) ЛЮБЫХ(ОЕ) ПОВТОРЯЮЩИХСЯ(АЕ) ЧЛЕНОВ(ЙЕ) ПОСЛЕДОВАТЕЛЬНОСТИ(КР) , АВТОРОМ(ЙТ) ДАН(УМ) АЛГОРИТМ(ЙИ), ДЛЯ(ПР) ОПРЕДЕЛЕНИЯ(ЛР) АВТОКОРРЕЛЯЦИОННОЙ(ЯЗ) СТРУКТУРЫ(КР) НОВОЙ(ЯЗ) ПОСЛЕДОВАТЕЛЬНОСТИ(КР).

 МОРФОЛОГИЯ

 В исследованиях по морфологии естественных языков, проведенных в последние годы отечественными лингвистами, можно выделить несколько

крупных направлений:

 — теоретические исследования по различным вопросам морфологии;

 — типологические исследования по морфологии;

 — прикладные аспекты морфологических исследований.

 Теоретические исследования по морфологии

 По сравнению с другими областями языкознания морфология по праву

считается наиболее исследованной областью. Поэтому в завершающее десяти-

летие нашего века закономерно появление целого ряда фундаментальных публикаций итогового характера, вышедших из-под пера известных ученых,

посвященных основополагающим понятиям, своего рода principia morphologia.

К ним относится:

 — системное описание всех основных разделов морфологии: морфемики, изучающей морфемную членимость слова, принципы вычленения

и разграничения морфем, их функционирование в языке; словообразования,

изучающего соотношения между морфемными структурами; словопроизводства

— образования новых слов путем использования морфемных возможностей язы-

ка [Земская, Немченко, Панов, Тимофеев, Тихонов];

 — системное описание диахронической морфологии индоевропейских

языков [Журавлев] и древненовгородского диалекта русского языка [Зализняк];

 — итоговое описание системы морфологии русского языка ХХ века:

морфемной модели и количественных данных о русских морфемах [Кузнецова],

активных процессов в словоизменении, словообразовании и словопроизводстве

(суффиксации, префиксации, аббревиации, производстве сложных слов, типич-

ных особенностях окказиональных и потенциальных слов) [Земская, 92, 95,

Русский язык в его функционировании, Русский язык конца ХХ века], морфологических словарей русского языка [Кузнецова, Тихонов];

 — морфологическая типология слова в разноструктурных языках, преж-

де всего в славянских [Широкова];

 — структурное моделирование морфологии естественных языков в тесной

связи с достижениями структурализма в других областях языкознания, прежде

всего в фонологии и синтаксисе [Бондарко, Демьянков].

 В теории морфологии традиционно большое место занимают исследова-

ния по аспектологии русского глагола: анализ совершенного вида в отрицатель-

ных предложениях [Акимова], фактическое и общефактическое значение вида,

семантика и прагматика несовершенного вида императива, таксономические

категории глаголов imperfectiva tantum [Падучева], видовая парность русских

глаголов [Черткова], проблема инварианта в семантике вида [Шатуновский],

употребление глаголов вторичной имперфективизации [Русский язык в его

функционировании].

 Среди других теоретических работ следует отметить :

 — исследования по аффиксации: изучение словообразовательного потенциала суффиксальных типов русских существительных [Каде], суффиксальной универбации и усечения в русском словообразовании

[Осипова Л.И.], возникновения новых аффиксов [Мамрак], сочетаемости пре-

фиксов в русском слове [Кузнецова], парадигматики и синтагматики русских

глагольных префиксов [Волохина и др.], суффиксоидов в современном английском языке [Бартков], словаря словообразовательных аффиксов в русском языке [Улуханов, 93];

 — исследования словообразовательной мотивированности и ее связи с

производностью [Улуханов,92], мотивационного отношения “имя-глагол”

[Осипова М.А.], типов полисемии в производном слове и границ словообразова-

тельного гнезда [Ширшов, 96] ;

 — работы по различным типам окказиональных способов словообразо- вания [Улуханов, 92];

 — исследования по связи морфологического уровня с другими уровнями

языка: фонемным (морфемный или фонемный принцип русской орфографии)

[Моисеев,95], синтаксическим (синтаксическая деривация как проявление изоморфизма между словообразованием и синтаксисом) [ ], лексическим

(лексический состав русского языка со словообразовательной точки зрения

[Моисеев, 91], семантическим (границы словообразовательной семантики) [Улуханов,91], стилистическим (стилистические аспекты русского словообразования ) [Vinogradova];

 — работы по диахронической морфологии русского словообразования [Улуханов, 92], диахронии словоизменения и словообразования русских существительных [Русский язык в его функционировании], развития категории

одушевленности в русском языке [Крысько].

 Как и в каждой области знания, в морфологии немало интересных и спор-

ных проблем. Поскольку “словообразование постоянно балансирует в языке между системностью и беспорядком” [Пиотровский, 95], в литературе живо обсуждаются правомерность использования понятия усечения морфем [Добродомов], сложные случаи морфемики и словообразования [Шанский], пограничные случаи между словоизменением и словообразованием [Муравьева], лексикой и словообразованием [Глухих], морфемика в ее отношении к формообразованию [Герд, 94], проблемы трактовки групповой флексии [Плун-

гян,94].

 С точки зрения историографии морфологии безусловный интерес представ-

ляет переписка Н.С.Трубецкого и Р.О.Якобсона о повелительном наклонении

русского глагола [Храковский, 94].

 Типологические исследования по морфологии

 Естественными и традиционными представляются работы по сопостави-

тельной морфологии русского языка с другими языками и прежде всего со сла-

вянскими, балканскими и балтийскими. Из огромного количества подобных

работ прежде всего следует отметить доклады российской делегации на X I Меж-

дународном съезде славистов в Братиславе (1993г.) [Земская и др., Нещименко,

 ], материалы научных конференций

и сборников, в которых среди прочих обсуждаются проблемы:

 — сопоставительной славянской аспектологии [Смирнов];

 — образно-экспрессивного употребления грамматических категорий [Ши-

рокова];

 — словообразовательной интерференции в славянских языках [Джамбазов];

 — морфологической типологии слова в разноструктурных языках, в том

числе в русском, английском, французском и испанском [Широкова,92];

 — типологии грамматических категорий — в частности, категории лишитель-

ности в славянских и балканских языках [Иванов и др.], категории наклонения

в разных языках [Сабанеева];

 — морфологические характеристики слов в “старославянском словаре” [Кры-

сько].

 Как всегда, весьма разнообразны сопоставительные морфологические исследования по другим языкам: английскому [Биренбаум], малагасийскому [Коршунов], тагальскому [Шкарбан], чешскому [Стешковская], эвенскому

[Роббек], лезгинскому [Керимов], эскимосскому [Вахтин], вепсскому [Иткин],

догон [Плунгян, Plungian], дари [Островский].

 Прикладные исследования по морфологии

 Прикладная морфология, являясь составной частью компьютерной линг-

вистики, традиционно считается в ней наиболее исследованной областью,

в задачу которой входит:

 — разработка морфологических процессоров — систем автоматического

морфологического анализа и синтеза слов, а также систем лемматизации — сведе-

ния словоформ к словарным словам;

 — автоматизация морфологических исследований, предусматривающая про-

ведение ряда лингвистических работ с помощью ЭВМ для решения основной за-

дачи — разработки морфологических процессоров: использование СУБД для

составления морфологических словарей, проведения типологических исследо-ваний, моделирования морфологических явлений и т.д.

 Следует отметить, что в настоящее время используются самые разнообраз-

ные автоматизированные системы обработки речи и текста, применяющие мор-

фологические процессоры: системы машинного перевода, коррекции и редакти-

рования текста, анализа и синтеза речи, информационного поиска (в особенности

полнотекстовые системы), автоматического реферирования, автоматические словари, экспертные системы и обучающие лингвистические автоматы. В указанных системах морфологические процессоры являются наиболее надеж-

ными и хорошо отработанными компонентами. В последних промышленных и коммерческих разработках хорошо зарекомендовали себя морфологические процессоры процедурного типа, работающие на базе словаря основ, — ср. сис-

темы автоматического морфологического анализа в СМП АСПЕРА [Королев,

91,95], MORSE [Пиотровский, 95], в автоматических корректорах WinОРФО [Ашманов, 95], автоматического морфологического синтеза в СМП СПРИНТ-2 [Тихомиров, 92], лемматизации русских слов [QUALICO-94].

 Модификация морфологических процессоров в основном идет за счет

использования блока анализа слов, не содержащихся в словаре (“новых слов”),

 которые традиционно анализируются на основе принципа морфологической аналогии, предполагающего сильную корреляционную зависимость между

грамматическими характеристиками слов и буквенным составом их концов.

Этот принцип давно и успешно используется в морфологических процессорах, разработанных Г.Г.Белоноговым. В 80-е годы киевскими лингвистами (В.И.Пе-

ребейнос, Т.А.Грязнухина, Н.П.Дарчук и др.) принцип аналогии был положен

в основу морфологических анализаторов русского языка, работающих без ка-

ких-либо словарей. Модификация этого подхода, также предполагающего ис-

ключение словарей для целей морфологического анализа или использование

их в минимальной степени, в последнее время была предпринята Г.Г.Белоно- говым для русского и некоторых других языков [Белоногов и др.,95], а также другими специалистами по вычислительной морфологии [Шереметьева и др., 96].

 Широкое распространение персональных ЭВМ создает благоприятные

условия для автоматизации морфологических исследований — автоматического

формирования русского морфологического словаря по исходным массивам слов

и словосочетаний [Большаков,93], использования специализированных словарных

баз данных для анализа морфологии русского языка, автоматического типологи-

ческого анализа морфологии семитских языков [QUALICO-94], автоматического

 формирования перечня структурных типов префиксов русских существительных

[Герд, 93], использования автоматизированной системы составления и ведения флективных классов русских существительных, прилагательных и глаголов, применяемых в морфологическом процессоре СМП АСПЕРА [Королев, 95],

моделирования морфологического анализа русских слов, содержащих суффик-

сы [Гельбух, 92].

 В заключение напомним о том, что за последние несколько десятилетий

морфология переживала свои периоды взлетов и падений. В 60-е годы нашего

столетия определенный застой в морфологических исследованиях был вызван

бурным распространением структурного синтаксиса, а затем и семантики. Одна-

ко со временем развитие науки и требования жизни все расставили по своим ме-

стам, и в последние годы, как свидетельствует наш краткий обзор, теоретическая

и прикладная морфология вновь являются важным полигоном для лингвистичес-

кой теории и практики.

Обеспечение взаимодействия с ЭВМ на естественном языке (ЕЯ) является важнейшей задачей исследований по искусственному интеллекту (ИИ). Базы данных, пакеты прикладных программ и экспертные системы, основанные на ИИ, требуют оснащения их гибким интерфейсом для многочисленных пользователей, не желающих общаться с компьютером на искусственном языке. В то время как многие фундаментальные проблемы в области обработки ЕЯ (Natural Language Processing, NLP) еще не решены, прикладные системы могут оснащаться интерфейсом, понимающем ЕЯ при определенных ограничениях.

Существуют два вида и, следовательно, две концепции обработки естественного языка:

для отдельных предложений;

для ведения интерактивного диалога.

Природа обработки естественного языка

Обработка естественного языка - это формулирование и исследование компьютерно-эффективных механизмов для обеспечения коммуникации с ЭВМ на ЕЯ. Объектами исследований являются:

собственно естественные языки;

использование ЕЯ как в коммуникации между людьми, так и в коммуникации человека с ЭВМ.

Задача исследований - создание компьютерно-эффективных моделей коммуникации на ЕЯ. Именно такая постановка задачи отличает NLP от задач традиционной лингвистики и других дисциплин, изучающих ЕЯ, и позволяет отнести ее к области ИИ. Проблемой NLP занимаются две дисциплины: лингвистика и когнитивная психология.

Традиционно лингвисты занимались созданием формальных, общих, структурных моделей ЕЯ, и поэтому отдавали предпочтение тем из них, которые позволяли извлекать как можно больше языковых закономерностей и делать обобщения. Практически никакого внимания не уделялось вопросу о пригодности моделей с точки зрения компьютерной эффективности их применения. Таким образом, оказалось, что лингвистические модели, характеризуя собственно язык, не рассматривали механизмы его порождения и распознавания. Хорошим примером тому служит порождающая грамматика Хомского, которая оказалась абсолютно непригодной на практике в качестве основы для компьютерного распознавания ЕЯ.

Задачей же когнитивной психологии является моделирование не структуры языка, а его использования. Специалисты в этой области также не придавали большого значения вопросу о компьютерной эффективности.

Различаются общая и прикладная NLP. Задачей общей NLP является разработка моделей использования языка человеком, являющихся при этом компьютерно-эффективными. Основой для этого является общее понимание текстов, как это подразумевается в работах Чарняка, Шенка, Карбонелла и др. Несомненно, общая NLP требует огромных знаний о реальном мире, и большая часть работ сосредоточена на представлении таких знаний и их применении при распознавании поступающего сообщения на ЕЯ. На сегодняшний день ИИ еще не достиг того уровня развития, когда для решения подобных задач в большом объеме использовались бы знания о реальном мире, и существующие системы можно называть лишь экспериментальными, поскольку они работают с ограниченным количеством тщательно отобранных шаблонов на ЕЯ.

Прикладная NLP занимается обычно не моделированием, а непосредственно возможностью коммуникации человека с ЭВМ на ЕЯ. В этом случае не так важно, как введенная фраза будет понята с точки зрения знаний о реальном мире, а важно извлечение информации о том, чем и как ЭВМ может быть полезной пользователю (примером может служить интерфейс экспертных систем). Кроме понимания ЕЯ, в таких системах важно также и распознавание ошибок и их коррекция.

Основная проблема обработки естественного языка

Основной проблемой NLP является языковая неоднозначность. Существуют разные виды неоднозначности:

Синтаксическая (структурная) неоднозначность: во фразе *Time flies like an arrow* для ЭВМ неясно, идет ли речь о времени, которое летит, или о насекомых, т.е. является ли слово *flies* глаголом или существительным.

Смысловая неоднозначность: во фразе *The man went to the bank to get some money and jumped in* слово *bank* может означать как *банк*, так и *берег*.

Падежная неоднозначность: предлог *in* в предложениях *He ran the mile in four minutes/He ran the mile in the Olympics* обозначает либо время, либо место, т.е. представлены совершенно различные отношения.

Референциальная неоднозначность: для системы, не обладающей знаниями о реальном мире, будет затруднительно определить, с каким словом - *table* или *cake* - соотносится местоимение *it* во фразе *I took the cake from the table and ate it*.

Литерация (Literalness): в диалоге *Can you open the door? — I feel cold* ни просьба, ни ответ выражены нестандартным способом. В других обстоятельствах на вопрос может быть получен прямой ответ *yes/no*, но в данном случае в вопросе имплицитно выражена просьба открыть дверь.

Центральная проблема как для общей, так и для прикладной NLP - разрешение такого рода неоднозначностей - решается с помощью перевода внешнего представления на ЕЯ в некую внутреннюю структуру. Для общей NLP такое превращение требует набора знаний о реальном мире. Так, для анализа фразы *Jack took the bread from the supermarket shelf, paid for it, and left* и для корректного ответа на такие вопросы, как *What did Jack pay for?*, *What did Jack leave?* и *Did Jack have the bread with him when he left?* необходимы знания о супермаркетах, процессах покупки и продажи и некоторые другие.

Прикладные системы NLP имеют преимущество перед общими, т.к. работают в узких предметных областях. К примеру, системе, используемой продавцами в магазинах по продаже компьютеров, не нужно ”раздумывать” над неоднозначностью слова *terminals* в вопросе *How many terminals are there in the order?*.

Тем не менее, создание систем, имеющих возможность общения на ЕЯ в широких областях, возможно, хотя пока результаты далеки от удовлетворительных.

Технологии анализа естественного языка

Под технологией анализа ЕЯ подразумевается перевод некоторого выражения на ЕЯ во внутреннее представление. Фактически все системы анализа ЕЯ могут быть распределены на следующие категории: подбор шаблона (Pattern Matching), синтаксический анализ, семантические грамматики, анализ с помощью падежных фреймов, “жди и смотри” (Wait And See), словарный экспертный (Word Expert), коннекционистский, “скользящий” (Skimming) анализ. Ниже пойдет речь о некоторых наиболее распространенных методах, описанных в статье.

Подбор шаблона. Сущность данного подхода состоит в интерпретации ввода в целом, а не в интерпретации смысла и структуры его отдельных составляющих на более низком уровне. При использовании этого метода происходит сравнение уже имеющихся в системе шаблонов-образцов с текстом, поступившим на вход. Обычно шаблоны представлены в виде простого списка соответствий между классами высказываний и интерпретациями. Иногда они дополнены семантическими элементами или другими компонентами более высокого уровня. По такому принципу работает система *Элиза*, имитирующая диалог с психотерапевтом. В действительности система ничего не понимает, а лишь поддерживает диалог, сравнивая реплики пациента с шаблонами и присвоенными им соответствующими ответными репликами, такими, как:

|  |  |
| --- | --- |
| *X всегда Y:* | Вы можете привести какой-нибудь пример? |
|  | Когда? |
|  | Действительно, всегда? |
| *X Вы Y меня:* | Почему Вы думаете, что я Y Вас? |
|  | Вам нравится думать, что я Y Вас, не правда ли? |
|  | Представьте, что я Y Вас. Что Вы об этом думаете? |

В результате *Элиза* способна вести такой диалог:

|  |  |  |
| --- | --- | --- |
| *Пац.:* | Люди всегда пристально смотрят на меня. | подбор по 1-му шаблону с учетом *всегда* |
| *Эл.:* | Вы можете привести какой-нибудь пример? | первый ответ по 1-му шаблону |
| *Пац.:* | Например, сейчас ты смотришь на меня. | подбор по 2-му шаблону, Y=*смотреть* |
| *Эл.:* | Почему Вы думаете, что я смотрю на Вас? | первый ответ по 2-му шаблону |
| *Пац.:* | Потому что ты ненавидишь меня. | подбор по 2-му шаблону, Y=*ненавидеть* |
| *Эл.:* | Вам нравится думать, что я ненавижу Вас, не правда ли? и т.д. | второй ответ по 2-му шаблону |

Синтаксический анализ. При использовании синтаксического анализа происходит интерпретация отдельных частей высказывания, а не всего высказывания в целом. Обычно сначала производится полный синтаксический анализ, а затем строится внутренне представление введенного текста, либо производится интерпретация.

*Деревья анализа и свободно-контекстные грамматики.* Большинство способов синтаксического анализа реализовано в виде деревьев. Одна из простейших разновидностей - свободно-контекстная грамматика, состоящая из правил типа *S=NP+VP* или *VP=V+NP* и полагающая, что левая часть правила может быть заменена на правую без учета контекста. Свободно-контекстная грамматика широко используется в машинных языках, и с ее помощью созданы высокоэффективные методы анализа. Недостаток этого метода - отсутствие запрета на грамматически неправильные фразы, где, например, подлежащее не согласовано со сказуемым в числе. Для решения этой проблемы необходимо наличие двух отдельных, параллельно работающих грамматик: одной - для единственного, другой - для множественного числа. Кроме того, необходима своя грамматика для пассивных предложений и т.д. Семантически неправильное предложение может породить огромное количество вариантов разбора, из которых один будет превращен в семантическую запись. Всё это делает количество правил огромным и, в свою очередь, свободно-контекстные грамматики непригодными для NLP.

*Трансформационная грамматика.* Трансформационная грамматика была создана с учетом упомянутых выше недостатков и более рационального использования правил ЕЯ, но оказалась непригодной для NLP. Трансформационная грамматика создавалась Хомским как порождающая, что, следовательно, делало очень затруднительным обратное действие, т.е. анализ.

*Расширенная сеть переходов.* Расширенная сеть переходов была разработана Бобровым (Bobrow), Фрейзером (Fraser) и во многом Вудсом (Woods) как продолжение идей синтаксического анализа и свободно-контекстных грамматик в частности. Она представляет собой узлы и направленные стрелки, “расширенные” (т.е. дополненные) рядом тестов (правил), на основании которых выбирается путь для дальнейшего анализа. Промежуточные результаты записываются в ячейки (регистры). Ниже приводится пример такой сети, позволяющей анализировать простые предложения всех типов (включая пассив), состоящие из подлежащего, сказуемого и прямого дополнения, таких, как *The rabbit nibbles the carrot (Кролик грызет морковь)*. Обозначения у стрелок означают номер теста, а также либо признаки, аналогичные применяемым в свободно-контекстных грамматиках (*NP*), либо конкретные слова (by). Тесты написаны на языке LISP и представляют собой правила типа *если условие=истина, то присвоить анализируемому слову признак Х и записать его в соответствующую ячейку.*

Разберем алгоритм работы сети на вышеприведенном примере. Анализ начинается слева, т. е. с первого слова в предложении. Словосочетание *the rabbit* проходит тест, который выясняет, что оно не является вспомогательным глаголом (*Aux*, стрелка 1), но является именной группой (*NP*, стрелка 2). Поэтому *the rabbit* кладется в ячейку *Subj,* и предложение получает признак *TypeDeclarative*, т.е. *повествовательное*, и система переходит ко второму узлу. Здесь дополнительный тест не требуется, поскольку он отсутствует в списке тестов, записанных на LISP. Следовательно, слово, стоящее после *the rabbit* - т. е. *nibbles* - глагол-сказуемое (обозначение *V* на стрелке), и *nibbles* записывается в ячейку с именем *V*. Перечеркнутый узел означает, что в нем анализ предложения может в принципе закончиться. Но в нашем примере имеется еще и дополнение *the carrot*, так что анализ продолжается по стрелке 6 (выбор между стрелками 5 и 6 осуществляется снова с помощью специального теста), и словосочетание *the carrot* кладется в ячейку с именем *Obj*. На этом анализ заканчивается (последний узел был бы использован в случае анализа такого пассивного предложения, как *The carrot was nibbled by the rabbit*). Таким образом, в результате заполнены регистры (ячейки) *Subj*, *Type*, *V* и *Obj*, используя которые, можно получить какое-либо представление (например, дерево).

Расширенная сеть переходов имеет свои недостатки:

немодульность;

сложность при модификации, вызывающая непредвиденные побочные эффекты;

хрупкость (когда единственная неграмматичность в предложении делает невозможным дальнейший правильный анализ);

неэффективность при переборе с возвратами, т.к. ошибки на промежуточных стадиях анализа не сохраняются;

неэффективность с точки зрения смысла, когда с помощью полученного синтаксического представления оказывается невозможным создать правильное семантическое представление.

Семантические грамматики. Анализ ЕЯ, основанный на использовании семантических грамматик, очень похож на синтаксический, с той разницей, что вместо синтаксических категорий используются семантические. Естественно, семантические грамматики работают в узких предметных областях. Примером служит система *Ladder*, встроенная в базу данных американских судов. Ее грамматика содержит записи типа:

S ® <present> the <attribute> of <ship>

<present> ® what is|[can you] tell me

<ship> ® the <shipname>|<classname> class ship

Такая грамматика позволяет анализировать такие запросы, как *Can you tell me the class of the Enterprise?* (*Enterprise* - название корабля). В данной системе анализатор составляет на основе запроса пользователя запрос на языке базы данных.

Недостатки семантических грамматик состоят в том, что, во-первых, необходима разработка отдельной грамматики для каждой предметной области, а во-вторых, они очень быстро увеличиваются в размерах. Способы исправления этих недостатков - использование синтаксического анализа перед семантическим, применение семантических грамматик только в рамках реляционных баз данных с абстрагированием от общеязыковых проблем и комбинация нескольких методов (включая собственно семантическую грамматику).

Анализ с помощью падежных фреймов. С созданием падежных фреймов связан большой скачок в развитии NLP. Они приобрели популярность после работы Филлмора “Дело о падеже”. На сегодняшний день падежные фреймы - один из наиболее часто используемых методов NLP, т.к. он является наиболее компьютерно-эффективным при анализе как снизу вверх (от составляющих к целому), так и сверху вниз (от целого к составляющим).

Падежный фрейм состоит из заголовка и набора ролей (падежей), связанных определенным образом с заголовком. Фрейм для компьютерного анализа отличается от обычного фрейма тем, что отношения между заголовком и ролями определяется семантически, а не синтаксически, т.к. в принципе одному и то же слово может приписываться разные роли, например, существительное может быть как инструментом действия, так и его объектом.

Общая структура фрейма такова:

[Заголовочный глагол

[падежный фрейм

*агент:* <активный агент, совершающий действие>

*объект:* <объект, над которым совершается действие>

*инструмент:* <инструмент, используемый при совершении действия>

*реципиент:* <получатель действия - часто косвенное дополнение>

*направление:* <цель (обычно физического) действия>

*место:* <место, где совершается действие>

*бенефициант:* <сущность, в интересах которой совершается действие>

*коагент:* <второй агент, помогающий совершать действие>

]]

Например, для фразы *Иван дал мяч Кате* падежный фрейм выглядит так:

[Давать

[падежный фрейм

*агент:* Иван

*объект:* мяч

*реципиент:* Катя]

[грам

*время:* прош

*залог:* акт]

]

Существуют обязательные, необязательные и запрещенные падежи. Так, для глагола *разбить* обязательным будет падеж *объект* - без него высказывание будет незаконченным. *Место* и *коагент* будут в данном примере необязательными падежами, а *направление* и *реципиент* - запрещенными.

Часто в NLP бывает полезным использовать семантическое представление в как можно более канонической форме. Наиболее известным способом такой репрезентации являются метод концептуальных зависимостей, разработанный Шенком для глаголов действия. Он заключается в том, что каждое действие представлено в виде одного или более простейших действий.

Например, для предложений *Иван дал мяч Кате (1)* и *Катя взяла мяч у Ивана (2),* различающихся синтаксически, но оба обозначающих акт передачи, могут быть построены следующие репрезентации с использованием простейшего действия *Atrans*, применяющегося в грамматике концептуальных зависимостей:

|  |  |
| --- | --- |
| (1) | (2) |
| [Atrans | [Atrans |
| *отн:* обладание | *отн:* обладание |
| *агент:* Иван | *агент:* Катя |
| *объект:* мяч | *объект:* мяч |
| *источник:* Иван | *источник:* Иван |
| *реципиент:* Катя] | *реципиент:* Катя] |

С помощью такого представления легко выявляются сходства и различия фраз.

Для облегчения анализа также используется деление роли на лексический маркер и заполнитель. Так, для роли *объект* может быть установлен маркер *прямое дополнение*, для роли *источник* - маркер вида *<маркер-из>=из|от|*...

В общем анализ текста с помощью падежных фреймов состоит из следующих шагов:

Используя существующие фреймы, подобрать подходящий для заголовка. Если такого нет, текст не может быть проанализирован.

Вернуть в систему подходящий фрейм с соответствующим заголовком-глаголом.

Попытаться провести анализ по всем обязательным падежам. Если один или более обязательных заполнителей падежей не найдены, вернуть в систему код ошибки. Такой случай может означать наличие эллипсиса, неверный выбор фрейма, неверно введенный текст или недостаток грамматики. Следующие шаги используются уже для анализа и исправления таких ситуаций.

Провести анализ по всем необязательным падежам.

Если после этого во введенном тексте остались непроанализированные элементы, выдать сообщение об ошибке, связанной с неправильным вводом, недостаточностью данного анализа или необходимостью провести другой, более гибкий анализ.

Преимущества использования падежных фреймов таковы:

совмещение двух стратегий анализа (сверху вниз и снизу вверх);

комбинирование синтаксиса и семантики;

удобство при использовании модульных программ.

Устойчивость анализа

Определенную трудность при анализе представляет вариативность одного и того же запроса. Например, на вход системы, управляющей зачислением и перераспределением учащихся на курсах разных специальностей, может поступить запрос типа *Переведите Петрова, если это возможно, с математики на, скажем, экономику*.

Наиболее легко такие трудности преодолеваются при использовании падежных фреймов. Правило, сформулированное Карбонеллом и Хейзом, гласит: “Следует пропускать неизвестные введенные элементы до тех пор, пока не будет найден падежный маркер; пропущенные элементы следует анализировать с учетом незаполненных падежей, используя только семантику”.

Диалог

Наряду с проблемой распознавания текста существует и проблема поддержания интерактивного диалога. При этом возникают дополнительные особенности, характерные для диалогов, а именно:

анафора (т.е. использование местоимений вместо их анафорических антецедентов - самостоятельных частей речи);

эллипсис;

экстраграмматические предложения (пропуск артиклей, опечатки, употребления междометий и т.п.);

металингвистические предложения (т.е. попытка исправления введенного ранее).

Кроме того, пользователи систем с естественно-языковым интерфейсом стараются выражаться как можно короче, что в ряде случаев также затрудняет анализ.

Использование падежных фреймов, а именно слияние текущего фрейма с предыдущим, обеспечивает восстановление эллипсиса.

Заключение

Таким образом, процесс разработки систем, обеспечивающих понимание ЕЯ, требует создание механизмов, отличных от традиционных способов представлений ЕЯ, а системы с естественно-языковыми интерфейсами применяются только в узких предметных областях.

АНАЛИЗ СНИЗУ ВВЕРХ И СВЕРХУ ВНИЗ

“Сверху вниз” vs. “снизу вверх”, “прямой” vs. “обратный”, “управляемый данными” vs. “движимый целью” - три пары определений для таких терминов, как “цепной анализ”, “парсинг”, “синтаксический разбор”, “логический анализ” и “поиск”. В принципе, все эти термины отражают сходные отношения, и различие между ними состоит лишь в том, что они взяты из различных подобластей компьютерной науки и искусственного интеллекта (парсинг, системы с заложенными в них правилами, поисковые системы и системы, направленные на решение проблем и т.д.)

Суть этих противопоставлений можно проиллюстрировать на примере парадигмы поиска. Основная задача любого поиска состоит в том, чтобы определить маршрут, по которому вы будете перемещаться с настоящей позиции к вашей цели. Если вы начнете поиск с текущей позиции и будете продолжать его, пока не наткнетесь на желаемый результат, - это так называемый прямой поиск или поиск снизу вверх. Если вы мысленно ставите себя в то место, где вы хотите очутиться в результате поиска и определяете маршрут, двигаясь в обратном направлении, т.е. туда, где вы действительно находитесь в настоящий момент, - это поиск в обратном направлении или поиск сверху вниз. Обратите внимание на то, что, определив маршрут в результате обратного поиска, вам все же предстоит добраться до своей цели. Несмотря на то, что сейчас вы движетесь вперед, это не является прямым поиском, т.к. поиск уже был осуществлен ранее, причем в обратном направлении.

Эти же противопоставления можно рассмотреть на примере систем с встроенными правилами. Представим себе, что правило состоит из набора антецедентов и набора следствий. Когда система определяет, что все антецеденты определенного правила удовлетворены, это правило вызывается и выполняется (выполняется ли каждое вызванное правило зависит от специфики конкретной системы). После этого в базу знаний заносятся утверждения, полученные в результате выполнения правила, и выполняются соответствующие операции. Данный процесс происходит вышеописанным образом, независимо от того, применяет ли система прямой или обратный логический анализ. Чтобы проиллюстрировать различия между ними, следует отдельно рассмотреть процедуру активации правила. Вызываются только активированные правила. При прямом логическом анализе (снизу вверх), когда в систему добавляются новые данные, они сравниваются со всеми антецедентами всех правил. Если данные соответствуют антецеденту правила, то это правило активируется (если оно еще не является активированным), и если подобраны все антецеденты определенного правила, то оно вызывается. Утверждения, полученные в результате выполнения правила, заносятся в базу знаний и рассматриваются в качестве новых данных, сравниваются с антецедентами и могут вызвать активацию и вызов дополнительных правил. При обратном логическом анализе (сверху вниз) при добавлении данных правила не активируются. Когда система получает запрос, он сравнивается со всеми следствиями всех правил. Если запрос совпадает со следствием, то это правило активируется, а все его антецеденты рассматриваются в качестве вторичных запросов и могут вызвать активацию дополнительных правил. Когда запрос соответствует не ограниченному условием утверждению базы знаний, на него поступает ответ, и если этот запрос исходил от антецедента, считается, что он удовлетворяет последнему. Когда все антецеденты некоторого правила будут удовлетворены, правило вызывается и выполняется. При выполнении правила осуществляется ответ на запросы, которые его активировали, и теперь другие антецеденты считаются удовлетворенными и могут вызываться соответствующие им правила. Обратите внимание на то, что вызов и выполнение правила всегда происходит в прямой последовательности, а отличие прямого цепного анализа от обратного состоит в том, когда активируется правило.

Примеры

Парсинг. Попытаемся проиллюстрировать и объяснить разницу между синтаксическим анализом сверху вниз и снизу вверх на примере предложения “They are flying planes” и простой грамматики, представленной в виде пронумерованных правил:

 1. S NP VP

 2. NP N

 3. NP PRO

 4. NP ADJ N

 5. VP VT NP

 6. VT V

 7. VT AUX V

 8. N planes

 9. PRO they

 10. ADJ flying

 11. AUX are

 12. V are

 13. V flying

Антецеденты указаны с правой стороны, а следствия - с левой. Например, правило 1 читается следующим образом: “Если последовательность состоит из именной группы (NP), за которой следует глагольная группа (VP), то эта последовательность является предложением (S).”

 Синтаксический разбор сверху вниз начинается с символа S, который и будет являться вершиной дерева разбора. Эта процедура эквивалентна процедуре постановки задачи, которая заключается в том, чтобы определить, является ли последовательность слов предложением. Правило 1 гласит, что каждое предложение состоит из именной группы (NP), за которой следует глагольная группа (VP). При наличии нескольких правил, сперва применяется правило с наименьшим номером, а затем оно расширяется слева направо. Таким образом следующим шагом является нахождение первой связи, т.е. NP. Сперва активируется правило 2, а затем правило 8 (рис. 2*а*). Т.к. “planes” не соответствует ”they”, алгоритм срабатывает вновь, и теперь сперва активируется правило 3, а затем правило 9. Затем алгоритм возвращается к правилу 1 и следующей целью ставит определение VP. Сперва активируются правила 5, 6, а затем 12 (рис. 2*b*). Дальнейший ход разбора отржен на рисунке 2 (*с, d, e).*

Синтаксический разбор снизу вверх начинается со слов в предложении. Опять же разбор ведется слева направо, и сперва применяется правило с наименьшим номером. Итак, сначала первое слово предложения “they” соотносится с антецедентом правила 9, которое после выполнения выдает утверждение, что “they” является местоимением (PRO). Затем выполняется правило 3 и выдает, что “they” является NP. NP соответствует антецедентам правил 1 и 5, но ни одно из этих правил еще не вызвано, поэтому разбор переходит к “are”. Выполняется правило 11 (несмотря на то, что правило 12 также вызвано, оно не выполняется в соответствии с правилом о последовательности выполнения правил). Затем выполняются правила 10, 8 и 2 (рис. 3*а*). На данной стадии дальнейший разбор последовательности NP+AUX+ADJ+NP невозможен, поэтому мы возвращаемся к последнему вызванному, но еще не выполненному правилу, т.е. к правилу 4. Разбор последовательности NP+AUX+NP так же невозможен, поэтому опять выполняется последнее вызванное невыполненное правило. Сейчас это правило 13, которое выдает, что “flying” является V. Затем выполняются правила 6 и 5 (рис. 3*с*).Разбор последователльностиNP+AUX+VP невозможен, поэтому выполняется правило 7 и выдает утверждение, что “are flying” является VT. Затем снова выполняются правила 5 и 1, на чем и заканчивается синтаксический разбор (рис. 3*d*).

Данный пример был приведен с целью сравнения механизмов синтаксического разбора снизу вверх и сверху вниз. Установление строгого порядка разбора слева направо и нумерация правил обусловлены стремлением к применению в наибольшей степени сходного алгоритма, несмотря на то, что результаты разбора оказались различными.

Системы со встроенными правилами. Рассмотрим прямой и обратный цепной анализ на примере выдуманного набора правил о том, как следует провести вечер. Правила расположены в обычном порядке, антецедент располагается слева, а следствие - справа, все вызванные правила выполняются, а разбор ведется параллельно.

1. Хороший фильм по ТВ + Рано утром встреч нет Позднее кино

2. Рано утром встреч нет + Нужно поработать Работа допоздна

3. Нужно поработать + Необходимы документы Работа в офисе

4. Позднее кино Не спать допоздна

5. Работа допоздна Не спать допоздна

6. Работа допоздна Возвращение в офис

7. Работа в офисе Возвращение в офис

Например, правило 1 гласит, что если по ТВ идет хорошее кино и у меня завтра рано утром встреч нет, тогда я следую режиму “Позднее кино”.

Рассмотрим сперва пример прямого цепного анализа. Допустим, система получила начальную информацию о том, что завтра рано уторм у меня нет встреч. Активируются правила 1 и 2. Допустим, что далее система получила сообщение о том, что мне нужно поработать. Активируется правило 3, а правило 2 вызывается и выполняетя, откуда следует вывод, что я нахожусь в режиме “Работа допоздна”, в результате чего вызываются и выполняются правила 5 и 6. В итоге система заключает, что я должна вернуться в офис и не спать допоздна.

Теперь рассмотрим эту же проблему с применением обратного цепного анализа. Допустим, что система получила исходную информацию о том, что у меня нет завтра утром встреч, но мне нужно еще поработать, а затем ее (систему) спросили, следует ли мне вернуться в офис. Данный запрос активирует правила 6 и 7. В свою очередь возникнет вопрос “Работа допоздна” или “Работа в офисе”? При этом активируются правила 2 и 3, и возникает вопрос “Рано утром встреч нет”, “Нужно поработать” или “Нужны документы”? Первые два антецедента будут удовлетворены, таким образом правило 2 будет вызвано и выполнено, что повлечет за собой удовлетворение антецедента “Работа допоздна”, вызов и выполнение правила 6, в результате чего система придет к заключению, что мне следует вернуться в офис.

Обратите внимание на то, что при прямом разборе порождается больше следствий, а при обратном - запросов. Т.к. в обоих примерах использовались одни и те же данные, то в ходе анализа выполнялись одни и те же правила, но активировались различные.

Сравнение

Эффективность. Выбор вида анализа (сверху вниз или снизу вверх) зависит от конфигурации дерева, по которому осуществляется поиск. Если в среднем каждому элементу *следует* большее количество элементов, нежели *предшествует*, то анализ сверху вниз (или обратный анализ) будет более эффективным и наоборот. Рассмотрим крайний случай. Допустим, что поисковая область образует дерево с вершиной в начальном состоянии. Тогда при использовании прямого подхода нам придется осуществлять поиск практически по всему дереву, тогда как при обратном подходе - только в его линейной части.

Сравнение и унификация. В системах с заложенными правилами или системах логического анализа выбор прямого или обратного цепного анализа влияет на степень трудности процесса сравнения. При прямом цепном анализе системе постоянно предъявляются новые факты, не имеющие свободных переменных. Таким образом постоянно проводится сравнение антецедентов, вполне вероятно обладающих свободными переменными, с фактами, не обладающими таковыми.

С другой стороны, системам с обратным цепным анализом често задают специальные вопросы. Если правила изложены в логике предикатов, а не логике суждений, тогда производится сравнение вопроса с переменной со следствием с переменными. Вторичные запросы также могут содержать переменные, поэтому, в общем, системы с обратным цепным анализом должны быть разработаны таким образом, чтобы они могли сравнивать две символьные структуры, каждая из которых может содержать переменные, для чего потребуется создание алгоритма унификации.

Смешанные стратегии

Поиск в двух направлениях. Если не ясно, какой вид поиска - прямой или обратный - является наиболее приемлимым для конкретного приложения, следует осуществлять поиск в двух направлениях. В таком случае, отправными точками становятся начальное и конечное состояние, и поиск осуществляется по направлению к центру.

Вывод по двум направлениям. При данном подходе изначальные данные применяются для активирования правил, котоые перебирают другие антецеденты в обратном порядке. Вторичные запросы, которые не соответствуют ни следствиям, ни данным, сохраняются в качестве “демонов”, которые могут быть удовлетворены позднее за счет новых или позднее поступивших данных. Систему можно разработать таким образом, что данные, удовлетворяющие “демонам” (антецеденты активированных правил) не будут активировать дополнительные правила, что “заставит” систему при предстоящем прямом выводе сконцентрироваться на правилах, учитывающих предыдущий контекст.

Разбор с началом в левом углу. Применив вышеописанный метод к парсингу, мы получим так называемый разбор с началом в левом углу. В терминах примера, приведенного в разделе парсинг, система сначала рассмотрит “they”, найдет правило 9 - единственное правило, которое можно применить к этому слову, затем правило 3, объясняющее PRO, а затем правило 1, как единственное правило, следствие которого начинается с NP. Далее система попытается разобрать сверху вниз “are flying planes” как VP.

Заключение

Обычно в системах искусственного интеллекта применяется один из двух видов анализа. Первый - это анализ снизу вверх или прямой анализ, а второй- сверху вниз или обратный. Различие их определяется тем, в каком направлении ведется поиск (от начала в конец или наоборот) и какой элемент (следствие или антецедент) активирует правила.

Фактор эффективности и легкости внедрения может сыграть решающую роль при выборе вида анализа, который будет применяться в определенном приложении, но следует помнить, что использование смешанных стратегий также возможно.

СЕМАНТИЧЕСКИЕ СЕТИ.

Семантическая сеть - структура для представления знаний в виде узлов, соединенных дугами. Самые первые семантические сети были разработаны в качестве языка-посредника для систем машинного перевода, а многие современные версии до сих пор сходны по своим характеристикам с естественным языком. Однако последние версии семантических сетей стали более мощными и гибкими и составляют конкуренцию фреймовым системам, логическому программированию и другим языкам представления.

Начиная с конца 50-ых годов были создано и применены на практике десятки вариантов семантических сетей. Несмотря на то, что терминология и их структура различаются, существуют сходства, присущие практически всем семантическим сетям:

1. узлы семантических сетей представляют собой концепты предметов, событий, состояний;

2. различные узлы одного концепта относятся к различным значениям, если они не помечено, что они относятся к одному концепту;

3. дуги семантических сетей создают отношения между узлами-концептами (пометки над дугами указывают на тип отношения);

4. некоторые отношения между концептами представляют собой лингвистические падежи, такие как агент, объект, реципиент и инструмент (другие означают временные, пространственные, логические отношения и отношения между отдельными предложениями;

5. концепты организованы по уровням в соответствии со степенью обобщенности так как, например, сущность, живое существо, животное, плотоядное,;

Однако существуют и различия: понятие значения с точки зрения философии; методы представления кванторов общности и существования и логических операторов; способы манипулирования сетями и правила вывода, терминология. Все это варьируется от автора к автору. Несмотря не некоторые различия, сети удобны для чтения и обработки компьютером, а также достаточно мощны, чтобы представить семантику естественного языка.

ИСТОРИЧЕСКАЯ СПРАВКА.

Фрег представил логические формулы в виде деревьев, которые однако мало напоминают современные семантические сети. Еще одним пионером стал Чарльз Сандерз Прис, который использовал графические записи в органической химии.

Он сформулировал правила выводы с использованием экзистенциональных графов.

В психологии Зельц использовал графы для представления наследственности некоторых характеристик в иерархии концептов. Научные изыскания Зельца имели огромное влияние на изучение тактики в шахматах, который в свою очередь повлиял на таких теоретиков, как Саймон и Ньюэлл.

Что касается лингвистики, то первым ученым, занимавшимся разработкой графических описаний, стал Теньер. Он использовал графическую запись для своей грамматики зависимостей. Теньер оказал огромное влияние на развитие лингвистики в Европе.

Впервые семантические сети были использованы в системах машинного перевода в конце 50-х - начале 60-х годов. Первая такая система, которую создала Мастерман, включала в себя 100 примитивных концептов таких, как, например, НАРОД, ВЕЩЬ, ДЕЛАТЬ, БЫТЬ. С помощью этих концептов она описала словарь объемом 15000 единиц, в котором также имелся механизм переноса характеристик с гипертипа на подтип. Некоторые системы машинного перевода базировались на корреляционных сетях Цеккато, которые представляли собой набор 56 различных отношений, некоторые из которых - падежные отношения, отношения подтипа, члена, части и целого. Он использовал сети, состоящие из концептов и отношений для руководства действиями парсера и разрешения неоднозначностей.

В системах искусственного интеллекта семантические сети используются для ответа на различные вопросы, изучение процессов обучения, запоминания и рассуждений. В конце 70-х сети получили широкое распространение. В 80-х годах границы между сетями, фреймовыми структурами и линейными формами записи постепенно стирались. Выразительная сила больше не является решающим аргументом в пользу выбора сетей или линейных форм записи, поскольку идеи записанные с помощью одной формы записи могут быть легко переведены в другую. И наоборот, особо важное значение получили второстепенные факторы, как читаемость, эффективность, неискусственность и теоретическая элегантность, также учитываются легкость введения в компьютер, редактирование и распечатка.

РЕЛЯЦИОННЫЕ ГРАФЫ.

Самые простые сети, которые используются в системах искусственного интеллекта, - реляционные графы. Они состоят из узлов, соединенных дугами. Каждый узел представляет собой понятие, а каждая дуга - отношения между различными понятиями. На рисунке 1 представлено предложение “Собака жадно гложет кость”. Четыре прямоугольника представляют понятия собаки, процесса гложения, кости и такой характеристики, как жадность. Надписи над дугами означают, что собака является агентов гложения, кость является объектом гложения, а жадность - это манера гложения.

Терминология, использующаяся в этой области различна. Чтобы добиться некоторой однородности, узлы, соединенные дугами, принято называть графами, а структуру, где имеется целое гнездо из узлов или где существуют отношения различного порядка между графами, называется сетью. Помимо терминологии, использующейся для пояснения, также различаются способы изображения. Некоторые используют кружки вместо прямоугольников; некоторые пишут типы отношений прямо над дугами, не заключая их в овалы; некоторые используют аббревиатуры, например О или А для обозначения агента или объекта; некоторые используют различные типы стрелок. На рисунке 2 изображен граф концептуальных зависимостей Шенка. <=> означает агента. INGEST (поглощать) - один из примитивов Шенка: ЕСТЬ - ПОГЛОЩАТЬ твердый объект; ПИТЬ - ПОГЛОЩАТЬ жидкий объект; ДЫШАТЬ - ПОГЛОЩАТЬ газообразный объект. Дополнительная стекла слева показывает, что кость переход из неуказанного места к собаке.

Поскольку довольно сложно ввести в компьютер некоторые диаграммы и при этом они занимают много места при печати, многие ученые записывают свои графы в более компактном варианте. Например, то же предложение Сова предложил записать в линейном виде с использованием некоторых элементов из рисунка 1:

[ЕСТЬ]-

 (AGNT) -> [СОБАКА]

 (OBJ) -> [КОСТЬ]

 (MANR) -> [ЖАДНОСТЬ]

В этом варианте записи квадратные скобки обозначают понятия, а круглые скобки содержат в себе названия отношений. Все линейные формы записи очень похожи на фреймовые структуры.

ГРАФЫ С ЦЕНТРОМ В ГЛАГОЛЕ.

Глаголы соединяются с группой существительного с использованием падежных отношений. Например, с предложении “Mary gave a book to Fred”, Mary агент давания, book объект этого процесса, а Fred реципиент глагола “давать”. Помимо падежных отношений в предложении в естественном языке также имеются средства для связи отдельных предложений. Такие отношения необходимы для следующего:

Союзы. Самый простой способ соединить предложения - это поставить между ними союз. Некоторые союзы, как например “и”, “или”, “если” обозначают логическую связь; некоторые, такие как “после того, как”, “когда”, “пока”, “с тех пор, как” и “потому что”, выражают временные отношения и причину.

Глаголы, требующие подчиненное предложение. Падежные фреймы многих глаголов требуют подчиненного предложения, являющегося обычно прямым дополнением. К такому типу относятся глаголы “говорить”, “считать”, “думать”, “знать”, “быть убежденным”, “угрожать”, “пытаться” и др.

Определители, относящиеся к целому предложению. Многие наречия и пропозиционные фразы относятся только к глаголу, но некоторые определяют целое предложение. Такие наречия, как “обычно”, “вероятно”, в большинстве случаев ставятся в начале предложения. А например, слово “однажды” определяет весь рассказ, следующий после него.

Модальные глаголы и времена. Такие глаголы, как “may”, “can”, “must”, “should”, “would” и “could” имеют модальное значение и относятся ко всему предложению, где они встречаются. Временное отношение может быть выражено как формой прошедшего времени глаголов, так и обстоятельствами “сейчас”, “завтра” или “однажды” и другими.

Связанный дискурс. Помимо отношений, выраженных в одном предложении, существуют также отношения более высокого порядка между отдельными предложениями рассказа или какого-либо другого повествования. Многие из них не выражены эксплицитно: временные отношения и следование аргументов может быть, например, имплицитно выражено порядком следования предложения друг за другом в тексте.

Именно потому, что глагол отводится такая важная роль в предложении, многие теория делают его своим центральным связующим звеном. Этот подход берет свое начало из Индо-Европейской языковой семьи, где модальность и временные отношения выражаются изменением глагольной формы. Рассмотрим следующий пример: “While a dog was eating a bone, a cat passed by unnoticed”. В этом предложении сообщено, что, когда предложение “While a dog was eating a bone” являлось истинным, второе предложение “A cat passed unnoticed” также является истинным. На рисунке 3 изображен граф с центром в глаголе. Союз “while” (WHL) соединяет узел PASS-BY с узлом EAT. На рисунке 3 показано, что собака является агентом незамечания (not noticing).

Графы с центром в глаголе - это реляционные графы, где глагол считается центральным звеном любого предложения. Маркеры времени и отношения пишутся прямо рядом с концептами, которые представляют глаголы. Графы концептуальных зависимостей Роджера Шенка также используют этот подход.

Несмотря на то, что графы с центром в глаголе довольно гибкие по своей структуре, они обладают рядом ограничений. Одно из них заключается в том, что они не проводят разграничение между определителями, которые относятся только к глаголу, и определителями, относящимися к предложению целиком. Рассмотрим следующие примеры:

The dog greedily ate the bone.

Greedily, the dog ate the bone.

Эти графы также плохо справляются с предложениями, находящимися внутри других предложений.

При работе с реляционными графами возникают проблемы с передачей всего многообразия временных отношений и отношений модальности. Несмотря на то, что многие учение используют эти графы для решения сложных проблем, они так до сих пор и не разработали общего метода для их разрешения. В выше приведенном примере пометка PAST должна относится ко всему предложению, которое говорит о том, что собака ест кость, а не только к глаголу EAT, поскольку очевидно, что кость позже была съедена собакой целиком. Также должно быть указано, что процесс прохождения кошки и процесс не замечания ее собакой происходили в одно и то же время.

ПРОПОЗИЦИОННЫЕ СЕТИ.

В пропозиционных сетях узлы представляют целые предложения. Эти узлы являются точками соприкосновения для отношений между отдельными предложениями связанного текста. С другой стороны они определяют время и модальность для всего контекста. Представленные ниже примеры иллюстрируют отношения, для записи которых необходимы пропозиционные узлы:

Sue thinks that Bob believes that a dog is eating a bone.

If a dog is eating a bone, it is unwise to try to take it away from him.

В первом предложении для глаголов “think” и “believe” целое предложение является дополнением: Боб считает, что “А dog is eating a bone”, то, что думает Сью представляет собой более сложное предложение-“Bob believes that a dog is eating a bone”. Такое гнездование предложений внутри других предложений может повторятся сколь угодно большое количество раз. Чтобы изобразить такое предложение, необходимо использовать пропозиционные узлы, которые содержат гнездящиеся графы. На рисунке 4 изображена пропозиционная сеть для этого предложения. Отметим, что (EXP) - experiencer, то есть тот кто испытывает, соединяет THINK с Сью, а BELIEVE с Бобом, однако EAT и DOG соединены между собой агентивным отношением (AGNT). Причиной разного типа отношений является тот факт, что думать и считать-это состояния, испытываемые людьми, а поедание-это действие осуществляемое агентом.

Во втором примере представлены два предложения, находящиеся в отношении условия. Антецедентом является предложение “А dog is eating a bone”, а консеквентом предложение “It is unwise to try to take it away from him”. Инфинитивы “to try” и “to take” указывают на другие, гнездящиеся предложения. На гнездящиеся предложения также указывает оборот “it is unwise”. Для этого предложения также необходимо указать соответствие между “it”, “him” и “bone” и “dog”. Связи соответствия обозначены пунктиром. Для формальной записи этого предложения также используются кванторы общности и существования и некоторые элементы логики.

Все реляционные графы и графы с центром в глаголе имеют много общего. Однако среди них существуют также и отличия:

1. Включение контекста или всего лишь его условное обозначение с отсылкой на схеме.

2. Строгое гнездование: один и тот же концепт может или не может встречаться в двух разных контекстах, ни один из которых не гнездиться в другом.

3. Указание связей соответствия. При перекрещивающемся контексте, то есть когда они один и тот же концепт встречается в двух разных контекстах, эти связи не указываются.

 Однако это всего лишь стилистические расхождения, которые не влияют существенно на логику построения.

ИЕРАРХИЯ ТИПОВ.

Иерархия типов и подтипов является стандартной характеристикой семантических сетей. Иерархия может включать сущности: ТАКСА<СОБАКА<ПЛОТОЯДНОЕ<ЖИВОТНОЕ<ЖИВОЕ СУЩЕСТВО<ФИЗИЧЕСКИЙ ОБЪЕКТ<СУЩНОСТЬ. Они также могут включать в себя события: ЖЕРТВОВАТЬ<ДАВАТЬ<ДЕЙСТВИЕ<СОБЫТИЕ или состояния: ЭКСТАЗ<СЧАСТЬЕ<ЭМОЦИОНАЛЬНОЕ СОТОЯНИЕ<СОСТОЯНИЕ. Иерархия Аристотеля включала в себя 10 основных категорий: субстанция, количество, качество, отношение, место, время, состояние, активность и пассивность. Некоторые учение дополнили его своими категориями.

Символ < между более общим и более частным символом читается как: “Х-тип/подтип У”.

Термин “иерархия” обычно обозначает частичное упорядочение, где одни типы являются более общими, чем другие. Упорядочение является частичным, потому, что многие типы просто не подлежат сравнению между собой. Сравним HOUSE<DOG и DOG<HOUSE бессмысленны, если их сравнивать, однако слово DOGHOUSE является подтипом HOUSE, но не DOG. Рассмотрим некоторые виды графов:

Ацикличный граф. Любое частичное упорядочение может быть изображено, как граф без циклов. Такой граф имеет ветви, которые расходятся и сходятся вместе опять, что позволяет некоторым узлам иметь несколько узлов-родителей. Иногда такой тип графа называют путанным.

Деревья. Самым распространенным видом иерархии является граф с одной вершиной. В такого рода графах налагаются ограничения на ацикличные графы: вершина графа представляет собой один общий тип, и каждый другой тип Х имеет лишь одного родителя У.

Решетка. В отличие от деревьев узлы в решетке могут иметь несколько узлов родителей. Однако здесь налагаются другие ограничения: любая пара типов Х и У как минимум должна иметь общий гипертип ХиУ и подтип ХилиУ. Вследствие этого ограничения решетка выглядит, как дерево, имеющее по главной вершине с каждого конца. Вместо всего одной вершины решетка имеет одну вершину, которая является гипертипом всех категорий, и другую вершину, которая является подтипом всех типов.

НАСЛЕДОВАНИЕ.

Основным свойством иерархии является возможность наследования подтипами качеств гипертипов: все характеристики, которые присущи ЖИВОТНОМУ, также присущи МЛЕКОПИТАЮЩЕМУСЯ, РЫБЕ и ПТИЦЕ. В основе теории наследования лежит теория силлогизмов Аристотеля: Если А - характеристика В, а В - х-ка С, то А хар-ка всех С.

Преимущества иерархии и наследования:

Иерархия типов является отличной структурой для индексирования базы знаний и ее эффективной организации.

Следование по какой-либо ветви с помощью иерархии осуществляется гораздо быстрее.

СИНТАКСИЧЕСКИЙ АНАЛИЗ ЯЗЫКА И ЕГО ПОРОЖДЕНИЕ.

Семантические сети могут помочь парсеру разрешить семантическую неоднозначность. Без такого рода представления вся тяжесть анализ языка падает на синтаксические правила и семантические тесты. Структура же семантической сети ясно показывает, как отдельные концепты соединены между собой. Когда парсер встречает какую-либо неоднозначность, он может использовать семантическую сеть для того, чтобы выбрать тот или иной вариант. При работе с семантическими сетями используется несколько техник парсинга.

Парсинг, в основе которого лежит синтаксис. Работа парсера контролируется грамматикой непосредственных составляющих и операторами построения структур и их тестирования. В то время, как данные на входе анализируются, операторы построения структур создают семантическую сеть, а операторы тестирования проверяют ограничения на частично построенной сети. Если никакие ограничения не найдены, то используемое при этом грамматическое правило отвергается и парсер проверяет другую возможность. Это самый распространенный подход.

Синтаксический анализатор с использованием семантики. Синтаксический анализатор с использованием семантики оперирует также как и парсер, в основе которого лежит синтаксис. Однако он оперирует не с синтаксическими категориями типа группа подлежащего и группа сказуемого, а с концептами высокого уровня типа КОРАБЛЬ и ПЕРЕВОЗИТЬ.

Концептуальный парсинг. Семантическая сеть предсказывает возможные ограничения, которые могут встретится в отношениях между словами, а также прогнозировать слова, которые позже могут встретиться в предложении. Например, глагол давать требует одушевленного агента и а также прогнозирует возможность реципиента и объекта, который будет дан. Шенк был одним из самых активных сторонников концептуального парсинга.

Парсинг, основанный на экспертизе слов. Вследствие существования большого количества неправильных образований в естественном языке, многие люди вместо того, чтобы обращаться к каким-либо универсальным обобщениям, используют специальные словари, представляющих собой совокупность некоторых независимых процедур, которые называются экспертами слов. Анализ предложения рассматривается как процесс, осуществляемый совместно различными словарными экспертами. Главным сторонником этого подхода был Смол.

Аргументы за и против различных техник парсинга часто основывался не на конкретные данные, а больше на уже устоявшемся мнении. И лишь один проект на практике сравнил несколько видов парсинга - это Язык Семантических Репрезентаций, проект разработанный в Университете Берлина. В течение нескольких лет они создали четыре разных вида парсеров для анализа немецкого языка и его записи на Язык Семантических Репрезентаций, который представляет собой сеть.

Первым парсером был парсер, созданный по подобию концептуального парсера Шенка. Было отмечено, что хотя добавление в его лексикон новых слов было довольно легко, анализ однако мог проводиться только на простых предложениях и только относительных придаточных. Расширить область синтаксической обработки этого парсера оказалось сложной задачей.

Второй парсер был семантически ориентированные расширенные сети перехода. В нем было легче обобщить синтаксис, однако аппарат синтаксиса работал медленнее, чем у первого рассмотренного парсера.

Затем работа велась с парсером словарных экспертов. Здесь легко велась обработка особых случаев, однако разбросанность грамматики между отдельными составляющими делала практически невозможным ее общее понимание, поддержку и модифицирование.

Парсер, который был создан относительно недавно, - это синтаксически ориентированный парсер, основанный на общей грамматике фразовой структуры. Он наиболее систематичен и обобщен и относительно быстр.

Эти результаты в принципе соответствуют мнению других лингвистов: синтаксически ориентированные парсеры наиболее целостны, однако для них необходим определенный набор сетевых операторов для плавного взаимодействия между грамматикой и семантическими сетями.

Порождение языка по семантической сети представляет собой обратный парсинг. Вместо синтаксического анализа некоторй цепочки с целью порождения сети генератор языка производит парсинг сети для получения некоторой цепочки. Существует два варианта порождения языка из семантической сети.

1. Генератор языка просто следует по сети, превращая концепты в слова, а отношения, указанные рядом с дугами, в отношения естественного языка. Этот метод имеет много ограничений.

2. Подходы, ориентированные на синтаксис контролируют порождение языка с помощью грамматических правил, которые используют сеть для того, чтобы определить, какое следующее правило нужно применить.

Однако на практике оба метода имеют много сходств: например, первый способ представляет собой последовательность узлов, которые обрабатываются генератором языка, ориентированным на синтаксис.

ОБУЧЕНИЕ МАШИН.

Графы и сети представляют собой простые понятия для программ, которые изучают новые структуры. Их преимущество при обучении заключается в легкости добавления и удаления, а также сравнения дуг и узлов. Ниже представлены программы, которые для обучения использовали семантические сети.

Винстон использовал реляционные графы для описания таких структур, как арки и башни. Машине предлагались примеры верного и неверного описания этих структур, а программа создавала графы, которые указывали все необходимые условия для того, чтобы эта структура была именно аркой или башней.

Салветер использовал графы с центром в глаголе для представления падежных отношений, которые требуют различные глаголы. Его программа MORAN для каждого глагола выведет падежный фрейм, сравнивая одни и те же ситуации до и после их описания с использованием этого глагола.

Шенк разработал теорию Memory-Organization Packets для объяснения того, как люди узнают новую информацию из конкретных жизненных ситуаций. При этом MOP-это это обобщенная абстрактная структура, которая не имеют отношения ни к одной конкретной ситуации в отдельности.

ПРИМЕНИЕ НА ПРАКТИКЕ.

Семантические сети могут быть записаны практически на любом языке программирования на любой машине. Самые популярные в этом отношении языки LISP и PROLOG. Однако многие версии были созданы и на FORTRANе, PASCALе, C и других языках программирования. Для хранения всех узлов и дуг необходима большая память, хотя первые системы были выполнены в 60-х годах на машинах, которые были гораздо меньше и медленнее современных компьютеров.

Один из самых распространенных языков, разработанных для записи естественного языка в виде сетей, - это PLNLP (Programming Language for Natural Language Processing) Язык Программирования для Обработки Естественного Языка, созданный Хайдерном. Этот язык используется для работы с большими грамматиками с обширным покрытием. PLNLP работает с двумя видами правил:

1. с помощью правил декодирования производится синтаксический анализ линейной языковой цепочки и строится сеть.

2. с помощью правил кодирования сканируется сеть порождается языковая цепочка или другая трансформированная сеть.

Помимо специальных языков для семантических сетей было также разработано специальное аппаратное обеспечение. На обычных компьютерах могут быть успешно выполнены операции с языками синтаксического анализа и операции сканирования сетей. Однако для больших баз знаний нахождение нужных правил или доступ к предзнаниям может потребоваться очень много времени. Чтобы позволить различным процессам поисках проходить одновременно Фальман разработал систему NETL, которая представляет собой семантическую сеть, которая может использоваться с параллельным аппаратным обеспечением. Таким образом он хотел создать модель человеческого мозга, в котором сигналы могут двигаться по различным каналам одновременно. Другие ученые разработали параллельное программное обеспечение для поиска наиболее вероятной интерпретации двусмысленных фраз естественного языка.

Синтез речи.

1 Ограничения на синтез речи.

Cуществуют различные методы синтеза речи. Выбор того или иного метода определяется различными ограничениями. Рассмотрим те 4 вида ограничений, которые влияют на выбор метода синтеза.

*Задача.*

Возможности синтезированной речи зависят от того, в какой области она будет применятся. Когда необходимо произносить ограниченное число фраз ( и их произнесение линейно не меняется ), необходимый речевой материал просто записывается на пленку. С другой стороны, если задача состоит в стимулировании познавательного процесса при чтении вслух, используется совершенно другой ряд методик.

*Голосовой аппарат человека.*

Все системы синтеза речи должны производить на выходе какую-то речевую волну, но это не произвольный сигнал. Чтобы получить речевую волну определенного качества, сигнал должен пройти путь от источника в речевом тракте, который возбуждает действие артикуляторных органов, которые действуют как изменяющиеся во времени фильтры. Артикуляторные органы также накладывают ограничения на скорость изменения сигнала. Они также имеют функцию сглаживания: гладкого сцепления отдельных базовых фонетических единиц в сложный речевой поток.

*Структура языка.*

Ряд возможных звуковых сочетаний опредляется природой той или иной языковой структуры. Было обнаружено, что еденицы и структуры, используемые лингвистами для описания и объяснения языка, могут также использоваться для характеристики и построения речевой волны. Таким образом, при построении выходной речевой волны используются основные фонологические законы, правила ударения, морфологические и синтаксические структуры, фонотактические ограничения.

*Технология.*

Возможности успешно моделировать и создавать устройства для синтеза речи в сильной степени зависят от состояния технико-технологической стороны дела. Речевая наука сделала большой шаг вперед благодаря появлению различных технолоний, в том числе: рентгенография, кинематография, теория фильтров и спектров, а главным образом - цифровые компьютеры. С приходом интегральных сетевых технологий с постоянно возрастающими возможностями стало возсожно построение мощных, компактных, недорогих устройств, действующих в реальном времени. Этот факт, вместе с основательными знаниями алгоритмов синтеза речи, стимулировал дальнейшее развитие систем синтеза речи и переход их в практическую жизнь, где они находят широкое применение.

2 Методы синтеза.

Различные подходы могут быть сгруппированы по областям их применения, по сложности их воплощения.

Синтезаторы делят на два типа: с ограниченным и неограниченным словарем. В устройствах с ограниченным словарем речь хранится в виде слов и предложений, которые выводятся в определенной последовательности при синтезе речевого сообщения. Речевые единицы, используемые в синтезаторах подобного типа, произносятся диктором заранее, а затем преобразуются в цифровую форму, что достигается с помощью различных методов кодирования, позволяющих компрессировать речевую информацию и хранить ее в памяти синтезирующего устройства. Существует несколько методов записи и компоновки речи.

*Волновой метод кодирования.*

Самый легкий путь - просто записать материал на пленку и по необходимости проигрывать. Этот способ обеспечивает высокое качество синтезируемой речи, т.к. позволяет воспроизводить форму естественного речевого сигнала. Однако этот путь синтеза не позволяет реализовать построение новой фразы, т.к. не предусматривает обращение к различным ячейкам памяти и вызов из памяти нужных слов. В зависимости от используемой технологии этот способ может представлять задержки в доступе и иметь ограничения, связанные с возможностями записи. Никаких знаний об устройстве речевого тракта и структуре языка не требуется. Единственно серьезное ограничение в данном случае имеет объем памяти. Существуют способы кодирования речевого сигнала в цифровой форме, позволяющие в несколько раз уплотнять информацию: простая модуляция данных, импульсно-кодовая модуляция, адаптивная дельтовая модуляция, адаптивное предиктивное кодирование. Данные способы могут уменьшить скорость передачи данных от 50кбит/сек (нормальный вариант) до 10кбит/сек, в то время как качество речи сохраняется. Естественно, сложность операций кодирования и декодирования увеличивается со снижением числа бит в секунду. Такие системы хороши, когда словарь сообщений небольшой и фиксированный. В случае же, когда требуется соединить сообщения в более длинное, сгенерировть высококачественную речь трудно, т.к. значения параметров речевой волны нельзя изменить, а они могут не подойти в новом контексте. Во всех системах синтеза речи устанавливается некоторый компромисс между качеством речи и гибкостью системы. Увеличение гибкости неизбежно ведет к усложнению вычислений.

*Параметрическое представление.*

С целью дальнейшего уменьшения требуемой памяти для хранения и обеспечения необходимой гибкости было разработано несколько способов, которые абстрагируются от речевой волны как таковой, а представляют ее в виде набора параметров. Эти параметры отражают наиболее характерную информацию либо во временной, либо в частотной области. Например, речевая волна может быть сформирована сложением отдельных гармоник заданной высоты и заданными спектральными выступами на данной частоте. Альтернативный путь состоит в том, чтобы форму речевого тракта описать в терминах акустики и искусственным путем создать набор резонансов. Этот метод синтеза экономичнее волнового, т.к. требует значительно меньшего объема памяти, но при этом он требует больше вычислений, чтобы воспроизвести исходный речевой сигнал. Данный способ дает возможность манипулировать теми параметрами, которые отвечают за качество речи (значение формант, ширина полос, частота основного тона, амплитуда сигнала). Это дает возможность склеивать сигналы, так что переходы на границах совершенно не заметны. Изменения таких параметров как частота основного тона на протяжении всего сообщения дают возможность существенно изменять интонацию и временные характеристики сообщения. Наиболее популярным в наст.вр. методами кодирования в устройствах, использующих параметрическое представление сигналов, является метод, основанный на формантных резонансах и метод линейного предсказания (LPC - linear predictive coding). Для синтеза используются единицы речи различной длины: параграфы, предложения, фразы, слова, слоги, полуслоги, дифоны. Чем меньше единица синтеза, тем меньшее их количество требуется для синтеза. При этом, требуется больше вычислений, и возникают трудности коартикуляции на стыках. Преимущества этого метода: гибкость, немного памяти для хранения исходного материала, сохранение индивидуальных характеристик диктора. Требуется соответствующая цифровая техника и знание моделей речеобразования, при этом, лингвистическая структура языка не используется.

*Синтез по правилам.*

Описанные выше методы синтеза ориентированы на такие речевые единицы, как слова, предварительно введенные в устройство с голоса диктора. Данный принцип лежит в основе функционирования синтезаторов с ограниченным словарем. В синтезаторах с неограниченным словарем элементами речи являются фонемы или слоги , поэтому в них применяется метод синтеза по правилам, а не простая компоновка. Данный метод весьма перспективен, т.к. обеспечивает работу с любым необходимым словарем, однако качество речи значительно ниже, чем при использовании метода компоновки.

При синтезе речи по правилам также используются волновой и параметрический методы кодирования, но уже на уровне слогов.

Метод параметрического представления требует компромисса между качеством речи и возможностью изменять параметры. Исследователи обнаружили, что для синтеза речи высокого качества необходимо иметь несколько различных произношений единицы синтеза (например, слога), что ведет к увеличению словаря исходных единиц без каких бы то ни было сведений о контекстной ситуации, оправдывающей тот или иной выбор. По этой причине процесс синтеза получает еще более абстрактный характер и переходит от параметрического представления к разработке набора правил, по которым вычисляются необходимые параметры на основе вводного фонетического описания.Это вводное представление содержит само по себе мало информации. Это обычно имена фонетических сегментов ( напр, гласные и согласные) со знаками ударения, обозначениями тона и временных характеристик. Таким образом, метод синтеза по правилам использует малоинформационное описание на входе ( менее 100 бит/сек). Этот метод дает полную свободу моделирования параметров, но необходимо подчеркнуть, что правила моделирования несовеншенны. Синтезированная речь хуже натуральной, тем не менее, она удовлетворяет тестам по разборчивости и понятности. На уровне предложения и параграфа правила предоставляют необходимую степень свободы для создания плавного речевого потока.

3 Конвертация текста в речь.

Синтез по правилам требует детального фонетического транскрибирования на входе. Хотя для запоминания этой информации требуется мало памяти, чтобы извлечь из нее необходимые параметры, необходимы знания эксперта. Для конвертации неограниченного английского текста в речь необходимо сначала проанализировать его с целью получения транскрипции, которая затем синтезируется в выходную речевую волну. Анализ текста по своей природе задача лингвистическая и включает в себя определение базовых фонетических, слоговых, морфемных и синтакисическмих форм, плюс - вычленение семантической и прагматической информации. Системы конвертации текста в речь являются наиболее комплексными системами синтеза речи, включающие в себя знания об устройстве речевого аппарата человека, лингвистической структуре языка, а также которые должны учитывать ограничения, накладываемые областью применения системы, технико-технологической базой. Необходимо заметить, что и текст и речь являются поверхностными представлениями базовых лингвистических форм, поэтому задача преобразования текста в речь состоит в выявлении этих базовых форм, а затем в воплощении их в речи.

4 Система преобразования текста в речь MITalk.

На примере этой системы проиллюстрируем сильные и слабые стороны коммерческих версий. Разработка системы началась в конце 60-х гг. Изначально предполагалось разработать читающую машину для слепых, но система MITalk может применяться в любых ситуациях, где необходимо преобразовать текст в речь. Система имеет блок морфологического анализа, правила преобразования буква-звук, правила лексического ударения, просодический и фонематический синтез.

5 Анализ текста

*Преобразование символов в стандартную форму.*

В самых различных текстах можно обнаружить символы и аббревиатуры, которые не принадлежат к категории " правильно образованных слов". Такие символы как "%" и "&", аббревиатуры типа "Mr" и "Nov" должны быть преобразованы в нормальную форму. Были разработаны подробные руководства по транскрибированию чисел, дат, сум денег. Иногда возникают двусмысленные ситуации, такие как, например, использование знака дефиса в конце строки. Человек в таких случаях, чтобы определить подходящее произношение, обращается к контексту и к практическим знаниям, которые не поддаются алгоритмизации.

*Морфологический анализ*

В вводном тексте границы слов легко определяются. Можно хранить произношение всех английских слов. Размер словаря будет большим, но в таком подходе есть несколько привлекательных сторон. Во-первых, в любом случае необходим словарь слов, произношение которых является исключением из общих правил. Такими являются, например, заимствованные слова ( parfait, tortilla). Более того, все механизмы преобразования цепочки букв в фонетические значки допускают ошибки. Интересный класс исключений составляют часто употребительные слова. Например, звук /th/ в начале слова произносится как глухой фрикативный в большинстве слов (thin, thesis, thimble). Но в наиболее частотных, таких как короткие функциональные слова the, this, there, these, those, etc. начальный звук произносится как звонкий. Также /f/ всегда произносится глухо, за исключением слова "of". Другой пример. В словах типа "shave", "behave" конечный /e/ удлиняет предшествующий гласный, но в таком частом слове как "have" это правило не действует. Наконец, конечный /s/ в "atlas", "canvas" глухой, но в функциональных словах is, was, has он произносится звонко. Таким образом, приходим к выводу, что все системы должны иметь такой словарь исключений. Что касается нормальных слов, то здесь имеется два варианта. Первый крайний случай состоит в том, чтобы составить полный словарь. Хотя число слов ограничено, составить абсолютно полный словарь невозможно, т.к. постоянно появляются новые слова. Кроме того, в словарь необходимо будет внести все изменяемые формы слова. Другой крайний подход состоит в установлении ряда правил, которые бы преобразовывали цепочки букв в фонетические значки. Хотя эти правила очень продуктивны, нельзя избежать ошибок, что ведет к созданию словаря исключений. Чтобы правильно определить фонетическую транскрипцию слова, нужно правильно разбить слово на структурные составляющие. Было обнаружено, что важную роль в определении произношения играет морфема, минимальная синтаксическая единица языка. Система MITalk использует морфемный лексикон, что может рассматриваться как некоторый компромиссный подход между двумя крайними, упомянутыми выше. Многие английские слова можно расчленить на последовательность морфов, таких как префиксы, корни, суффиксы. Так слово "snowplows" имеет два корня и окончание, "relearn" имеет приставку и корень. Такие морфы являются атомными составляющими слова и они относительно стабильны в языке, новые морфы формируются в языке очень редко. Эффективный лексикон может иметь не более 10,000 морфов. Морфемный словарь действует вместе с процедурами анализа. Этот подход эффективен и экономичен, т.к. хранение морфемного словаря не занимает много места, а хранить все изменяемые формы слова не нужно. Так как морфы являются основными составляющими слова, проиллюстрируем их полезность при определении произношения. При соединении морфов они часто меняют свое произношение. Например, при образовании множественного числа существительных "dog" и "cat" конечный /s/ будет звонким в первом случае и глухим во втором. Это пример морфофонемного правила, касающегося реализации морфемы множественного числа в различных окружениях. Становится очевидным, что для эффективного и легкого определения произношения нужно распознать составляющие морфемы слова и обозначить их границы. Еще один плюс морфемного анализа - обеспечение подходящей базы для использования правил преобразования буква-звук. Большинство таких правил рассматривают слово как неструктурированную последовательность букв, используя окно сканирования для нахождения согласных и гласных кластеров, которые преобразуются в фонетические значки. Буквы "t" и "h" в большинстве случаев выступают как единый согласный кластер, но в слове "hothouse" кластер /th/ разрывается границей двух разных морфем. Гласный кластер /ea/ представляет много трудностей для алгоритмов буква-звук, но в слове changeable он явно разрывается. В системе MITalk морфемный анализ всегда проводится перед правилами преобразования букв в звуки. Лежащие в основе слова морфы не всегда очевидны. Например, некоторые морфы множественного числа не всегда легко определить: mice, fish. Подобные формы заносятся в словарь. При помощи морфемного лексикона и соответствующего алгоритма анализа 95-98% слов анализируется удовлетворительно. В результате им приписывается фонетическая транскрипция и часть речи.

*Правила "буква-звук" и лексическое ударение*

В системе MITalk нормализованный вводный текст подвергается морфологическому анализу. Может быть, что целое слово есть в словаре морфов, как, например, слово "snow". С другой стороны, слово может быть проанализировано как последовательность соединенных морфов. В английском языке среднее число морфов в слове, примерно два. В случае, если ни целое слово не может быть найдено в словаре морфов, ни проанализировано как последовательность морфов, в этом случае применяются правила преобразования "буква-звук". Важно подчеркнуть, что этот метод никогда не применяется, если морфемный анализ удался. Конвертация последовательности букв в последовательность звуков при помощи этих правил проходит в три этапа. Первый этап - отделение префиксов и суффиксов. Возможность отделения аффиксов не такая сильная, как в морфемном анализе, но действует удовлетворительно. Предполагается, что после отделения префиксов и суффиксов остается одна центральная часть слова, которая состоит из одного морфа, подвергаемого затем правилам преобразования.

Второй этап состоит в преобразовании согласных в фонетические значки, начиная с наиболее длинного согласного кластера до тех пор, пока все отдельные согласные не будут преобразованы. Последний этап - оставшиеся гласные преобразуются при помощи контекстов. Гласные преобразуются последними, потому что это наиболее трудная задача, зависящая от контекста. Например, гласный кластер /ea/ имеет 14 разных произносительных контекстов и несколько произношений (reach, tear, steak, leather).

В системе MITalk правила преобразования букв в звуки действуют в паре с широким набором правил расстановки лексического ударения. Еще 25 лет назад лингвистам не удавалось обнаружить никакой системы расстановки ударений в английских словах. В Настоящее время разработан ряд правил, эффективно справляющихся с этой задачей. Ударения зависят от синтаксической роли слова, например, прилагательное "invalid" отличается от существительного. Таких слов немного, но учитывать их необходимо. Кроме того, на некоторые суффиксы автоматически падают ударения в словах, как, например, в "engineer". Но бывают более сложные случаи, которые разрешаются применением циклических правил.

В системе MITalk разработаны несколько наборов таких правил, некоторые из которых включают в себя до 600 правил. Конечно, большинство из них употребляются довольно редко. Подразумеваются, что все сильные и неправильные формы преобразуются на стадии морфологического анализа. Правила же "буква-звук" используются для преобразования новых и неправильно написанных слов. Например, слово "recieved" получает правильную транскрипцию, благодаря этим правилам преобразования.

*Парсинг.*

Каждая схема преобразования неограниченного текста в речь должна включать синтаксический анализ. Необходимо определить синтаксическую роль слова, т.к. она часто влияет на произношение и ударение. Кроме того синтаксический анализ важен для определения правильного тонального контура и временных характеристик. Просодические характеристики важны для синтеза речи, чтобы она звучала живо и естественно. К сожалению, полный синтаксический анализ на уровне сложного предложения (clause-level parsing) осуществить нельзя. Тем не менее, возможно провести синтаксический анализ на уровне фразы (phrase-level parsing), в результате которого определяется большая часть необходимой для синтеза речи структуры, хотя в некоторых ситуациях неизбежны ошибки из-за отсутсвия анализа целого предложения. Встречается множество синтаксически двусмысленных предложений, таких как "he saw the man in the park with a telescope", для которых фразовый анализ достаточен.

В английском языке существует ряд синтагматических маркеров, по которым можно формально разграничить фразы: это вспомогательные глаголы, детерминативы в номинативных фразах. Система MITalk широко использует это и проводит высокоточный грамматический анализ (augmented-transition-network grammas). Фразовый анализ показал удовлетворительные результаты, хотя эффективный анализатор предложений несомненно улучшил бы работу системы. Пока анализаторы предложений сталкиваются со значительными трудностями, когда встречают неполное или синтаксически омонимичное предложение. По завершении деятельности блока синтаксического анализа система приписывает словам маркеры функциональных частей речи, отмечает синтаксические паузы как основу для дальнейшего уточнения произношения, временных харатеристик, частоты основного тона.

*Модификация ударения и фонологические уточнения.*

Последняя фаза анализа состоит в некоторых незначительных поправках к имеющейся уже фонетической транскрипции на основе анализа контекстного окружения. Простой пример определения произношения артикля "the", которое зависит от начального звука последующего слова. Кроме того, на этом этапе используются некоторые эвристические методы проверки правильного соотношения общего контура предложения с контурами отдельных слов. На этом этапе заканчивается подготовка исходного текста собственно к самому процессу синтеза.

6 Синтез.

Важно осознать, что в системе MITalk не используются готовые речевые волны даже в параметрическом представлении. Система не хранит параметрические представления множества морфов или слов. Вместо этого были разработаны правила контроля параметров, так что можно реализовать любую желаемую речевую волну на выходе.

*Просодическая рамка.*

Первый шаг в создании выходной речевой волны - создание временного контура и частоты основного тона ( основные корреляты интонации ), на основе которых строится детальная артикуляция отдельных фонетических элементов. Распределение ударения, которое было вычислено на стадии анализа, во многом ответственно за контур временного распределения и тональный контур. Часто интенсивность принимают за коррелят ударения, тогда как главными ключами являются длительность и изменения в тональном контуре. Согласные мало меняются по длительности, в то время как гласные более пластичны и могут легко сжиматься или растягиваться. Существует также тенденция растягивать слова на границе основных абзацев предложения, и наоборот, сжимать интервалы на относительно невыделенных участках. Кроме того, на основе временной рамки задается частота основного тона (или тональный контур). В утвердительных предложениях обычно высота тона резко поднимается на первом ударном слоге, затем плавно снижается до последнего ударного слога, где она резко падает. Вопросительные и повелительные предложения имеют различные тональные контуры. Кроме целостного контура предложения существуют еще локальные ударения. Большее ударение получают слова, выражающие отрицание или сомнение ( например, слово might ), значение частоты основного тона на них возрастает; новая информация в предложении также больше выделяется ударением. С другой стороны, высота тона используется в семантических и эмоциональных целях, что не может быть выведено из письменного текста. Необходимо лишний раз подчеркнуть важность составления правильного просодического контура, т.к. неправильный просодический контур может привести к трудностям в восприятии.

*Синтез фонетических сегментов.*

Когда завершено создание просодической рамки, создаются параметры, соответствующие модели речевого тракта. Обычно таких параметров 25, которые изменяются с интервалом 5 - 10 мсек. В настоящее время используются около 100 контекстных правил описания траектории изменения параметров. Когда значения параметров вычислены, они должны быть перенесены на соответствующую модель речевого тракта (обычно это формантная модель или LPC-модель). Выходная дискретная модель создается обычно на частоте 10 Кгц.

7 Оценка синтетической речи.

С точки зрения понятности, разборчивости качество синтезированной речи достаточно хорошее. Был проведен тест, где одна группа испытуемых прослушивала синтезированную речь с письменным вариантом перед глазами, а другая - без. Выяснилось, что результаты прослушивания мало отличаются друг от друга. Тем не менее, синтезированной речи не хватает живости и естественности, поэтому воспринимать ее на протяжении длительного времени трудно. Исследования показали, что фрикативные и назальные звуки требуют дальнейшего улучшения качества.

Теория фреймов

*- это* парадигма для представления знаний с целью использования этих знаний компьютером . Впервые была представлена Минским как попытка построить фреймовую сеть , или парадигму с целью достижения большего эффекта понимания . С одной стороны Минский пытался сконструировать базу данных , содержащую энциклопедические знания , но с другой стороны , он хотел создать наиболее описывающую базу , содержащую информацию в структурированной и упорядоченной форме . Эта структура позволила бы компьютеру вводить информацию в более гибкой форме , имея доступ к тому разделу , который требуется в данный момент . Минский разработал такую схему , в которой информация содержится в специальных ячейках , называемых фреймами , объединенными в сеть , называемую системой фреймов . Новый фрейм активизируется с наступлением новой ситуации . Отличительной его чертой является то , что он одновременно содержит большой объем знаний и в то же время является достаточно гибким для того , чтобы быть использованным как отдельный элемент БД . Термин “фрейм” был наиболее популярен в середине семидесятых годов , когда существовало много его толкований , отличных от интерпретации Минского .

Чтобы лучше понять эту теорию , рассмотрим один из примеров Минского , основанный на связи между ожиданием , ощущением и чувством человека , когда он открывает дверь и входит в комнату . Предположим , что вы собираетесь открыть дверь и зайти в комнату незнакомого вам дома . Находясь в доме , перед тем как открыть дверь , у вас имеются определенные представления о том , что вы увидите , войдя в комнату . Например , если вы увидите к-л пейзаж или морской берег , поначалу вы с трудом узнаете их . Затем вы будете удивлены , и в конце концов дезориентированы , так как вы не сможете объяснить поступившую информацию и связать ее с теми представлениями , которые у вас имелись до того . Также у вас возникнут затруднения с тем , чтобы предсказать дальнейший ход событий. С аналитической точки зрения это можно объяснить как активизацию фрейма комнаты в момент открывания двери и его ведущую роль в интерпретации поступающей информации . Если бы вы увидели за дверью кровать , то фрейм комнаты приобрел бы более узкую форму и превратился бы во фрей кровати . Другими словами , вы бы имели доступ к наиболее специфичному фрейму из всех доступных .Возможно ,б что вы используете информацию , содержащуюся в вашем фрейме комнаты для того чтобы распознать мебель , что называется процессом сверху-вниз , или в контексте теории фреймов фреймодвижущим распознаванием . Если бы вы увидели пожарный гидрант , то ваши ощущения были бы аналогичны первому случаю. Психологи подметили , что распознавание объектов легче проходит в обычном контексте, чем в нестандартной обстановке . Из этого примера мы видим , что фрейм - это модель знаний , которая активизируется в определенной ситуации и служит для ее объяснения и предсказания . У Минского имелись достаточно расплывчатые идеи о самой структуре такой БД , которая могла бы выполнять подобные вещи . Он предложил систему , состоящую из связанных между собой фреймов , многие из которых состоят из одинаковых подкомпонентов , объединенных в сеть . Таким образом , в случае , когда к-л входит в дом , его ожидания контролируются операциями , входящими в сеть системы фреймов . В рассмотренном выше случае мы имеем дело с фреймовой системой для дома , и с подсистемами для двери и комнаты . Активизированные фреймы с дополнительной информацией в БД о том , что вы открываете дверь , будут служить переходом от активизированного фрейма двери к фрейму комнаты . При этом фреймы двери и комнаты будут иметь одинаковую подструктуру . Минский назвал это явление разделом терминалов и считал его важной частью теории фреймов .

Минский также ввел терминологию , которая могла бы использоваться при изучении этой теории ( фреймы , слоты , терминалы и т. д.) . Хотя примеры этой теории были разделены на языковые и перцептуальные , и Минский рассматривал их как имеющих общую природу , в языке имеется более широкая сфера ее применения . В основном большинство исследований было сделано в контексте общеупотребительной лексики и литературного языка .

Как наиболее доступную иллюстрацию распознаванию , интерпретации и предположению можно рассмотреть две последовательности предложений , взятых из Шранка и Абельсона . На глобальном уровне последовательность А явно отличается от В .

A John went to a restaurant

 He asked the waitress for a hamburger

 He paid the tip & left

B John went to a park

 He asked the midget for a mouse

 He picked up the box & left

Хотя все эти предложения имеют одинаковую синтаксическую структуру и тип семантической информации , понимание их кардинально различается . Последовательность А имеет доступ к некоторому виду структуры знаний высшего уровня , а В не имеет . Если бы А не имело такой доступ , то ее понимание сводилось бы к уровню В и характеризовалось бы как дезориентированное . Этот контраст является наглядным примером мгновенной работы высшего уровня структуры знаний .

Была предложена программа под названием SAM , которая отвечает на вопросы и выдает содержание таких рассказов . Например , SAM может ответить на следующие вопросы , ответы на которые не даны в тексте , с помощью доступа к записи предполагаемых событий , предшествующих обеду в ресторане .

Did John sit down in the restaurant ?

Did John eat the hamburger ?

Таким образом , SAM может распознать описанную ситуацию как обед в ресторане и затем предсказать оптимальное развитие событий . В нашем случае распознавание не представляло трудностей , но в большинстве случаев оно довольно непростое и является самой важной частью теории .

Рассмотрим другой пример :

C He plunked down $5 at the window .

 She tried to give him $ 2.50 , but he wouldn’t take it .

 So when they got inside , she bought him a large bag of popcorn .

Он интересен тем , что у большинства людей он вызывает цикл повторяющихся неправильных или незаконченных распознаваний и реинтерпретаций .

В случаях с многозначными словами многозначность разрешается с помощью активизированного ранее фрейма . Для этих целей необходимо создать лексикон к каждому фрейму . Когда фрейм активизируется , соответствующему лексикону отдается предпочтение при поиске соответствующего значения слова . В контексте ТФ это распознавание процессов , контролируемых фреймами , которые , в свою очередь , контролируют распознавание входящей информации . Иногда это называется процессом сверху - вниз фреймодвижущего распознавания .

Применение этих процессов нашло свое отражение в программе FRAMP , которая может суммировать газетные сводки и классифицировать их в соответствие с классом событий , например терроризм или землетрясения . Эта программа хранит набор объектов , которые должны быть описаны в каждой разновидности текстов , и этот набор помогает процессу распознавания описываемых событий .

 Манипуляция фреймами

Детали спецификации Ф и их репрезентации могут быть опущены , так же как и алгоритмы их манипуляции , потому что они не играют большой роли в ТФ .

Такие вопросы , как размер Ф или доступ к нему , связаны с организацией памяти и не требуют специального рассмотрения .

Распознавание

В литературе имеется много рассуждений по поводу процессов , касающихся распознавания фреймов и доступа к структуре знаний высшего уровня . Несмотря на то , что люди могут распознать фрейм без особых усилий , для компьютера в большинстве случаев это довольно сложная задача . Поэтому вопросы распознавания фреймов остаются открытыми и трудными для решения с помощью ИИ .

Размер фрейма

Размер фрейма гораздо более тесно связан с организацией памяти , чем это кажется на первый взгляд . Это происходит потому , что в понимании человека размер фрейма определяется не столько семантическим контекстом , но и многими другими факторами . Рассмотрим фрейм визита к доктору , который складывается из подфреймов , одним из которых является комната ожидания . Таким образом мы можем сказать , что размер фрейма не зависит от семантического содержания представленного фрейма / такого , как , например , визит к врачу / , но зависит от того , какие компоненты описывающей информации во фрейме / таком , как комната ожидания / используются в памяти . Это означает , что когда определенный набор знаний используется памятью более чем в одной ситуации , система памяти определяет это , затем модифицирует эту информацию во фрейм , и реструктурирует исходный фрейм так , чтобы новый фрей использовался как его подкомпонент .

Вышеперечисленные операции также остаются открытыми вопросами в ТФ .

Инициализационные категории

Рош предложил три уровня категорий представления знаний : базовую , субординатную и суперординационную . Например в сфере меблировки концепция кресла является примером категории основного уровня , а концепция мебели - это пример суперординационной категории . Язык представления знаний подвержен влиянию этой таксономии и включает их как различные типы данных . В сфере человеческого общения категории основного уровня являются первейшими категориями , которые узнают человек , другие же категории вытекают из них . То есть суперординационная категория - это обобщение базовой , а субординатная - это подраздел базовой категории .

 пример

 суперординатная идеи события

 базовая события действия

 субординатная действия прогулка

 Каждый фрейм имеет свой определенный так называемый слот . Так , для фрейма *действие* слот может быть заполнен только к-л исполнителем этого действия , а соседние фреймы могут наследовать этот слот .

Некоторые исследователи предположили , что случаи грамматики падежей совпадают со слотами в ТФ , и эта теория была названа теорией идентичности слота и падежа . Было предложено число таких падежей , от 8 до 20 , но точное число не определено . Но если агентив полностью совпадает со своим слотом , то остальные падежи вызвали споры . И до сих пор точно не установлено , сколько всего существует падежей .

Также вызвал трудность тот факт , что слоты не всегда могут быть переходными . Например , в соответствие с ТФ можно сказать , что фрейм *одушевленный предмет* может иметь слот *живой* , фрейм *человек* может иметь слот *честный* , а фрейм *блоха* не может иметь такой слот , и он к нему никогда не перейдет .

Другими словами , связи между слотами в ТФ не являются исследованными до конца . Слоты могут передаваться , могут быть многофункциональны , но в то же время не рассматриваются как функции . Гибридные системы

СФ иногда адаптируются для построения описаний или определений . Был создан смешанный язык , названный KRYPTON , состоящий из фреймовых компонентов и компонентов предикатных исчислений , помогающих делать к-л выводы с помощью терминов и предикатов . Когда активизируется фрейм , факты становятся доступными пользователю . Также существует язык Loops , который объединяет объекты , логическое программирование и процедуры .

Существуют также фреймоподобные языки , которые за исходную позицию принимают один тип данных в памяти , к-л концепцию , а не две / напр фрейм и слот / , и представление этой концепции в памяти должно быть цельным .

Объектно - ориентированные языки

Параллельно с языками фреймов существуют объектно - ориентированные программные языки , которые используются для составления программ , но имеют некоторые св-ва языков фреймов , такие , как использование слотов для детальной , доскональной классификации объектов . Отличие их от языков фреймов в том , что фреймовые языки направлены на более обобщенное представление информации об объекте .

 Одной из трудностей представления знаний и языка фреймов является отсутствие формальной семантики . Это затрудняет сравнение свойств представления знаний различных языков фреймов , а также полное логическое объяснение языка фреймов .

Трансформационная грамматика

Трансформационная грамматика - это одна из теорий описания естественного языка, основанная на предположении, что весь диапазон предложений любого языка может быть описан путем осуществления определенных изменений, или трансформаций, над неким набором базовых предложений. Разработанная Наумом Хомским (Noam Chomsky) в начале 50-х гг. и получившая свое развитие в ранних работах Зелига Харриса (Zellig Harris), теория трансформационной грамматики в настоящее время является чуть ли не единственной широко изучаемой и применяемой лингвистической моделью в США. В то же время необходимо отметить, что, в связи с возможностью по-разному трактовать большинство центральных идей данной теории, внутри нее в настоящий момент существует несколько соперничающих версий, претендующих на “правильную” интерпретацию трансформационной грамматики. Иногда трансформационную грамматику также называют генеративной грамматикой.

Синтаксические и семантические правила

Центральная идея трансформационной теории состоит в том, что поверхностные формы любого языка - его предложения - являются результатом взаимодействия между несколькими модульными подсистемами. Большинство версий трансформационной грамматики предполагают, что две базовые подсистемы из их общего числа - это набор синтаксических правил (ограничений) и набор семантических правил. Синтаксические правила определяют правильное расположение слов в предложениях (например, предложение “John will eat the ice cream” правильно, поскольку состоит из именной группы “John” и следующей за ним глагольной группы, или предиката, “will eat the ice cream”). Семантические правила отвечают за то, чтобы правильно интерпретировать конкретное расположение слов в предложении (например, “Will John eat the ice cream” является вопросом).

Синтаксические правила можно далее разделить на базовую грамматику, которая генерирует набор базовых предложений, и трансформационные правила, которые позволяют на основе базовых предложений создать производные предложения, или поверхностные структуры. Также существует дополнительный набор правил, которые на основе поверхностных структур создают произносимые выходные предложения.

Трансформационные правила

Трансформационные правила предназначены для описания систематических отношений в предложении, как то:

отличия между активным и пассивным предложением

глобальные отношения в предложении (например, связь между what и eat в предложении “What will John eat”)

неоднозначности, причиной которых является одна и та же форма предложения, выведенная из двух различных базовых предложений (например, в предложении “They are flying planes” flying можно рассматривать и как прилагательное и как основной глагол)

Базовое предложение “John will eat the ice-cream” может быть сгенерировано простым набором синтаксических правил, а затем, применив к нему трансформационные правила, можно построить производный вопрос “Will John eat the ice-cream”. С помощью другой последовательности трансформационных правил можно построить пассивное предложение: “Will the ice-cream be eaten by John”. В последнем случае мы видим, что в предложение добавились новые элементы be и by, а также изменились местоположение и форма старых элементов предложения.

Базовая грамматика

Базовые синтаксические признаки описываются грамматикой непосредственных составляющих, в простейшем случае контекстно-независимой грамматикой. Данная грамматика имеет следующий набор правил:

1) S → NP Aux VP 2) VP → Verb NP

3) NP → Name 4) NP → Determiner Noun

Auxiliary → will 6) Verb → eat

7) Determiner → the 8) Noun → ice cream

9) Name → John

Первое правило гласит, что предложение (S) - это именная группа (NP), за которой следует вспомогательный глагол (Aux) и затем глагольная группа (VP). Стрелку можно интерпретировать, как выражение “является” либо как команду “заменить символ S последовательностью NP Aux VP”. Подобным образом, второе правило гласит, что глагольная группа состоит из глагола, за которым следует именная группа. Третье и четвертое правило рассматривают именную группу, как имя собственное либо как существительное с детерминантом (определяемым словом). Последние пять правил являются лексическими; они вводят реальные слова, например, “”.

Символы типа “ice cream” называются терминальными элементами, так как они никогда не присутствуют в левой части правил. К ним нельзя далее применять никакие правила; на них как бы заканчиваются все действия правил. Все остальные символы, такие как S, NP, VP, Name и другие, считаются нетерминальными.

Все правила этой грамматики называются контекстно-независимыми, поскольку они позволяют свободно замещать любой символ слева от стрелки любой последовательностью символов справа от стрелки. С формальной точки зрения, контекстно-независимые правила имеют только один неразложимый символ, как то S, NP или VP, слева от стрелки.

Для того, чтобы сгенерировать базовый синтаксический признак, необходимо применить правила грамматики, начиная с символа S и до тех пор, пока никакие правила уже нельзя применить. Этот процесс называется деривацией, поскольку из символа S выводится новая цепочка символов. Результатом деривационного процесса может служить следующая запись:

Как правило, системы правил, подобные вышеописанной, подвергаются расширению с целью исключить возможность генерации бессмыслицы, типа “The ice cream ate” или “John took”. Для этого вводятся так называемые контекстно-зависимые правила, которые определяют контекст, дающий право заменять нетерминальные символы на терминальные. Например, символ V может быть заменен глаголом “took” только в том случае, если справа от него находится объект NP. Еще один пример: глагол “eat” может употребляться только после одушевленного существительного, что и должны подчеркивать контекстно-зависимые правила. Необходимо отметить, что в стандартной трансформационной теории 1965 года контекстно-зависимые лексические правила являлись частью словаря, а не базовой грамматики. В дополнение к лексическим контекстно-зависимым правилам, словарь содержит набор импликаций типа: “Если слово является именем человека, то оно также является одушевленным существительным.”

Словарь, состоящий из лексических ограничений и правил импликации, в сочетании с правилами базовой грамматики позволяет генерировать определенный набор базовых предложений. Ранее они назывались глубинными структурами, однако потом такая терминология была признана неудачной: данные формы не являются глубинными ни в том смысле, что они являются наиболее простыми и неразложимыми, ни в том смысле, что их значение является более глубоким; вследствие этого было решено отказаться от данной терминологии.

Трансформационный компонент

В соответствии с блок-схемой, базовые структуры далее поступают в трансформационный компонент, где для генерации дополнительных предложений могут применяться от нуля до нескольких трансформаций; на выходе этой процедуры получается поверхностная структура, которую уже можно произносить, как обычное предложение. Если не применяется ни одно из трансформационных правил, то поверхностная структура получается такой же, как и базовое предложение. Такое обычно происходит с простыми повествовательными предложениями, например:. Если же трансформационные правила все же применяются, то они производят новые синтаксические признаки, например: “Will John eat the ice-cream”.

Примером трансформационного правила может служить преобразование, создающее вопросительное предложение из синтаксического признака, который можно записать как X wh Y, где X и Y - любые цепочки символов в синтаксических признаках, а wh - - любая фраза, начинающаяся с wh, например, “who”, “what” или “what ice cream”. Цель этого трансформационного правила - переместить элемент wh в начало предложения. Если взять синтаксический признак, соответствующий предложению “John will eat what”, то его часть, соответствующая “John will eat” будет равна X, “what” - wh, а пустая последовательность - Y. Можно сделать вывод, что данная трансформация может иметь место. Переместив фразу с wh в начало, мы получим “What John will eat”. Применив к получившемуся синтаксическому признаку дополнительную трансформацию, а именно инверсию подлежащее - вспомогательный глагол, можно получить вопрос “What will John eat”. Необходимо отметить, что трансформационные правила применимы только к целым предложениям.

Традиционно, структурные описания и структурные изменения записываются путем присвоения элементам правила порядковых номеров и соответствующей записи. В нашем случае правило wh будет записано следующим образом:

Структурное описание: (X,wh,Y)

 (1,2,3)

Структурное изменение: (2,3,1)

Понимание речи

Понимание речи обычно трактуют как преобразование акустического представления речи в смысловое. При создании практических систем смысл можно определить, как представление, из которого извлекаются действия, совершенные системой. Понимание речи следует отличать от распознования речи, где целью является сопоставить речевое высказывание с соответствующими словами в словаре. До начала 70-ых большинство исследований было направлено на распознование речи. 5 лет потребовалось на создание системы ARPA, первоначальная исследовательская цель которой заключалась в распознавании речи, а конечные результаты в понимании. Казалось, что способность системы давать разумный ответ на речь была более значимым критерием для развития речевых систем. К тому же считалось, что речевой сигнал является недостаточным источником информации, и знание контекста речевого высказывания важно только для успешного распонавания и интерпретации. Системы по распознованию речи, основанные на динамическом программировании и соответствии с образцами, развивали для речевых высказываний, которые состояли почти полностью из изолированных слов, выбираемых из небольшого вокабуляра. Однако такой подход, при котором ищется наиболее точное соответствие между определенными произнесенными словами и вокабуляром акустическох образцов слов, меньше всего подходил к связанной речи, так как входной акустической сигнал в этом случае не может быть эффективно смоделирован, как простое сочетание произнесенных частей лексических единиц. В связанной речи изменчивость, выявляемая при соответствии с образцами, передает полезную информацию и для распознования, и для интерпретации. Однако, необходимо начинать с основных лингвистических единиц, таких как фонемы, и сохранять информацию о ритме и длительности речевого высказывания. Если следуют таким путем, то подход к обработке речи, основанный скорее на знании, чем на соответствиях с образцами, становится неизбежным, так как, чтобы извлекать преимущества из распознавания конкретных лингвистических единиц в сигнале, необходимо знать, как данная единица связана с остальной частью языка.

Системы понимания речи (СПР) имеют дело со связанными единицами речи, такими как, фразы, предложения и даже параграфы, так как "понимание" изолированных слов может означать только тривиальный процесс сопоставления некоторого значения к каждому слову словаря системы. Понимание связанной речи - очень сложная задача, и на проект СПР повлияли исследования в таких разных областях, как акустическая обработка сигнала, нейро-физиология, психолингвистика, психология. СПР была создана, чтобы понимать всего нескольких дикторов одного диалекта, производя грамматически ограниченное подмножество языка со словарем около тысячи слов. Сейчас хотя и имеются много потенциальных прикладных программ для СПР их эффективность и надежность все еще недостаточна, чтобы широко использоваться. Системы, зависимые от диктора, распознающие изолированные слова с небольшим словарем, использующие в качестве образцов-соответствий целые слова уже нашли свое применение, типа обработки багажа на авиалиниях. Тем не менее признано, что усовершенствование такого типа систем (большие словари, независимость от диктора) требует подхода, основанного на более глубоких знаниях.

Теоретические предпосылки

Посредником при преобразовании речи в ее значение должны служить определенные компоненты, которые используют разнообразные источники знания (ИЗ), т.к. речевой сигнал кодирует много различной информации, необходимой для восстановления значения. Например, вариативность в произношении слов в связанной речи больше не является помехой при подборе образца соответствия, но это довольно важный источник информации, например, относительно расположения границ слова или контекстуально важной (выделенной ударением) информации в произнесении. Единственной возможной организацией СПР и основных ИЗ является следующая: РЕЧЬ - ОБРАБОРТКА АКУСТИЧЕСКОГО СИГНАЛА - ФОНЕТИЧЕСКИЙ АНАЛИЗ - ФОНОЛОГИЧЕСКИЙ АНАЛИЗ - МОРФОЛОГИЧЕСКИЙ АНАЛИЗ - ЛЕКСИЧЕСКИЙ ДОСТУП К СЛОВАРЮ - СИНТАКСИЧЕСКИЙ АНАЛИЗ - СЕМАНТИЧЕСКИЙ АНАЛИЗ - ЗНАЧЕНИЕ. При такой организации СПР информация течет вверх по мере того, как каждый элемент создает промежуточные представления, кодируя (частичные) гипотезы относительно ввода на основе ему доступного знания.

Акустическая обработка отцифровывает сигнал с входной частотой, которая сохраняет сигнал для понимания. Акустическая обработка также трансформирует отцифрованный сигнал различными способами, чтобы представить его в той форме, которая поддается фонетическому декодированию. Например, спектральный анализ будет выполнен для каждого проанализированного фрейма, и дополнительные параметры, такие как частота основного тона, подсчитаны. Параметрический сигнал может затем быть помечен как дискретная последовательность фонем. Например, если сигнал с низкой амплитудой равномерно распространяется поперек спектра, то этот звук вероятно фрикативный, типа [f] или [v]. Кроме того, для каждой фонемы характерны такие особенности, как высота тона, длительность и амплитуда. Акустическо - фонетическое преобразование является решающим для эффективной работы СПР, но все еще одно из наиболее слабых сторон речевой обработки. И это являлось главным недостатком СПР, разработанной на основе ARPA в 1970-ых.

Фонологический анализ выполняется на фонетическом представлении, которое определяет лингвистически важные различия, имеющиеся в фонетическом представлении произнесения, например, уровни и расположение ударения, интонационный контур, структуры слога, последовательности фонем, лежащих в основе произнесения. Фонологический анализ необходим для лексического доступа, т.е. процесса, который сопоставляет фонетическую форму произнесения с каноническими фонемными представлениями слов в словаре, чтобы восстановить информацию, хранящуюся там относительно их морфологических, синтаксических, и семантических свойств. Это отменяет такие эффекты быстрой речи, как ассимиляция или сокращения. Например, слова “did” и "you" могли бы иметь в словаре следующие последовательности фонем: /dld/ и /ju:/. Однако, акустическо - фонетическое преобразование могло бы восстанавливать фактические звуки или фонемы, типа [dIje]; связывать эту фонетическую последовательность c каноническими фонемными представлениями “did” и "you". Это необходимо, если нужно узнать, что палатализация произошла на границе слова, заменив [dj] на [j], и что неударный гласный "you" был редуцирован до нейтрального безударного. Аналогично, фонологическое знание относительно допустимых последовательностей фонем в слогах может использоваться, чтобы распознать слог, и следовательно, границы слова. Например, в /houmhelp/ должна быть граница между /m/ и вторым /h/, потому что никакой слог в английском не может содержать /mh/.

Как только фонологический анализ завершен, дальнейшая обработка ввода будет подобна пониманию текста. Дальнейшие морфологический, синтаксический, семантический и прагматический анализы способствуют распознаванию, эксплуатируя избыточность речи, в информационно - теоретическом смысле. В некоторых из проектов APRA задача синтаксического анализа заключалась в том, чтобы исключить гипотезы слова на основе синтаксически недопустимых последовательностей.

Прежде, чем слова, выделенные в речевом сигнале будут сопоставлены с лексическими входам в словаре системы, необходимо провести морфологический анализ, который приведет слова к их основной форме, например, устранит окончание множественного числа /s/ или /z/, которые сильно бы расширили число входов в словарь.

После морфологического анализа возникшее морфофонологическое представление речевого ввода может быть найдено в словаре системы, чтобы получить синтаксическую и семантическую информацию относительно гипотезы последовательности слов. Синтаксический, семантический, и прагматический анализ - в основном тот же самый для речевого и текстового понимания. Однако, должно быть взаимодействие между этими и более низкими уровнями анализа не только, потому что они будут дополнять правильное распознавание произнесения, но также потому что некоторые аспекты фонологического анализа, особенно касающиеся ударения и интонации, будут способствовать интерпретации. Ударение, например, необходимо для определения контекстуально новой информации и для нахождению зависимых слов для местоимений.

Это краткое описание вклада различных ИЗ в понимание речи только раскрывает основные процессы. ИЗ, использованные в понимании речи, являются прежде всего лингвистическими. Однако, эффективность СПР зависит во много как от эффективного использования этих ИЗ так и от разработки их содержания.

Акустическо - фонетический Анализ

Несомненно наиболее важная область в обработке речи, нуждающаяся в исследованиях, - это акустическо - фонетический анализ. Если акустическо - фонетический анализ слабый, то ошибочные гипотезы выдадут в итоге неправильный анализ. Сегментация и идентификация акустического сигнала в последовательности лингвистических единиц чрезвычайно трудна. Сначала, речь - это код, а не шифр; то есть, акустическое сигналы, ассоциирующиеся с сегментами, непосредственно с ними не связанны; на эти сигналы сильно влияют соседние сегменты. Например, спектрограммы /d/ в /di/ и /du/ очень различны, т.к. на них влияют последующий гласный. Кроме того, не возможно разделить акустической сигнал на /d/ и следующий гласный. Эти наблюдения создали следующую теорию: конечное количество этих сегментов не всегда можно достичь из-за непрерывного движения вокального трактата. Такой синтезирующий анализ был бы, однако, очень в вычислительном отношении дорогой, так как он требовал бы, чтобы СПР умел генерировать всех возможные произнесения и сопоставлять их с акустическом вводом. Однако во-первых, акустическое сигналы, в противоположность фонемам или алафонам, содержат инвариантные сигналы. Во-вторых, акустическое сигналы часто сильно редуцируются в безударном положении. Это часто вызывает много неправильных гипотез в системах, где акустическо - фонетический компонент будет принимать за гипотезу сегмент из фиксированного инвентаря. В-третьих, акустическое сигналы варьируют от диктора диктору из-за физиологических особенностей вокального тракта, различия в характеристиках речи и т.д.. Люди способны компенсировать эти различия быстро и плавно, но все еще мало понятно, как сделать этот процесс автоматическим. Большинство коммерческих систем распознавания речи требует длинного обучения, повторяя за пользователем каждое слово в словаре системы несколько раз и - следовательно очень зависимо диктора. В ARPA несколько из разработанных СПР достигли определенной степени независимости от диктора, пытаясь ввести параметр в акустическо - фонетический анализ для нового диктора на основе обучающегося предложения, которое знала система, пользователю же следовало его проговорить.

Во всех ARPA проектируют СПР, где акустическо - фонетический анализ фактически не существовал и сегментный анализ не был точным. Конечное представление каждой системы было главным образом определено эффективностью более высоких уровней анализа при исправлении ошибок на фонетическом уровне. Более современные системы используют более сложный акустическо - фонетический анализ, интегрируя информацию из ряда преобразований акустического сигнала и создавая несколько типов фонетических представлений, но эффективность все еще ограничивается в среднем 70% успешным распознаванием фонем из речевого высказывания, произнесенных небольшим количеством дикторов.

Фонологический Анализ

 Фонологический компонент необходим для любой, обрабатывающей речь, системы, основанной на знаниях, потому что система требует знания относительно фонологических процессов, активных в языке и в прикладных программах, чтобы восстанавливать канонические произношение слов, которые могут быть сопоставлены с соответствующими входами словаря, и получать дальнейшие сигналы к синтаксической и семантической/прагматической интерпретации речевого высказывания. Фонологические компоненты были разработаны для СПР и других систем ARPA. Однако, они были в значительной степени ограничены лексическими, сегментными процессами и обычно имели дело с фонологически управляемыми изменениями, генерируя альтернативное произношение для индивидуальных лексических единиц и сохраняя их в дополнительном словаре. Этот подход не может иметь дело адекватно с фонологическими процессами, которые соединяют границы слова, типа палатализации. Самая большая область прикладной программы для фонологического правила - интонационная фраза; следовательно, фонологию нельзя рассматривать в терминах различного произношения для лексических единиц. Фонологический анализ обеспечивает много важной информации для СПР; например, различные виды фонологического правила блокированы различными лингвистическими границами между сегментами. Полезно разложить на слоги и слова речь, сегментация может также обеспечить сведения для синтаксического анализа; палатализация соединяет границы слова, но блокирована на границах главных синтаксических составляющих, так что ее отсутствие может использоваться, чтобы решить неоднозначность относительно присутствия такой границы в данном месте речевого сигнала. Фонологические правила также изменяются среди диалектов. Следовательно, СПР, способные к пониманию дикторов с различными диалектами, требовали бы знания относительно этих различий и способности реконфигурировать себя для их речи. Палатализация, например, происходит чаще в американских диалектах, чем в британских или английских.

В конце семидесятых стали развиваться новые подходы к фонологии, такие как автосегментная, метрическая зависимости, фонология зависимости, для которых центральным является сверхсегментальный аспект. Некоторые из этих достижений были включены в СПР.

Интерпретация, основанная на источнике знаний

ИЗ бесполезны в СПР, если знание, которое они кодируют, не может быть представлено таким образом, который позволяет интерпретацию с помощью машины. Например, специалисты по фонетики обычно используют Международный Фонетический Алфавит для фонетической записи. Однако, так как выбор представления воздействует на прикладную программу знания, системы представления ИЗ в СПР часто являлись компромиссом между описательной адекватностью и вычислительной эффективностью. Например, в ARPA проектируют каждый СПР, используя идею синтаксического представления, чтобы не выражать все грамматические возможности английского языка. Формальный язык и теория автоматов предлагают эффективные алгоритмы для прикладной программы ИЗ, выраженные в наборах правил с соответствующими формальными свойствами. Например, минимально увеличенные контекстно - свободные записи для адекватного описания английского синтаксиса и фонологии. Однако, успехи этого вида не ведут автоматически в вычислительном отношении к ИЗ, так как наборы правил, требуемые, чтобы выразить знание в этой форме могут быть чрезвычайно большие. Кроме того, кажется маловероятно, что все ИЗ, используемые в СПР могут быть выражены внутри таких ограниченных записей. Тем не менее, более специализированные и мощные методы также были разработаны, типа интерпретаторов для промышленных систем или увеличенные сети переходов. Появляются некоторые экспертные оболочки системы, являющееся многообещающими прикладными программами для акустическо - фонетического преобразования. Чем лучше понимание специфической области, тем больше возможность представления знания адекватно и эффективно. Кроме того, вероятно, что различные схемы представления будут наиболее эффективны для различных ИЗ; следовательно, структура СПР, которая навязывает, одинаковую схему для всех ИЗ, типа HAERSAY-11 или HARPY, не идеальна.

На выбор представления воздействуют факторы, другие чем доступность методики интерпретации для специфической схемы; например, несколько СПР не пытаются отображать непосредственно между акустическом сигналом и фонетическим алфавитом, но создавать промежуточные представления, отмечая акустическо яркие особенности типа назальности, помогать процессу распознавания фонем. На представления также воздействует порядок, в котором расположены различные ИЗ, относящиеся к речевому сигналу и полной структуре СПР. Недавно было предложено, чтобы начальный фонетический анализ отмечал согласные, гласные, а также ударные и безударные слоги и что это простое представление должно использоваться, чтобы получить набор слов-кандидатов из соответственно организованного словаря. Детализированный фонетический анализ затем применялся бы к безударному слогу(слогам), чтобы распознать его между кандидатами.

Структура Системы

Большая часть литературы по СПР касается межкомпонентной связи во время обработки. Эта проблема является основной, т.к. неоднозначности должны быть решены быстро, чтобы избежать ненужного вычисления, и также потому, что избыточность между ИЗ может использоваться, чтобы разложить на множители неправильные гипотезы, вызванные или ошибками системы или подлинной неоднозначностью в речевом сигнале. Например, акустическо - фонетический компонент мог бы предложить аспирированный /p/ или /b/, за которым следует гласные и /t/, результатом этого предположения могут стать такие слова-кандидаты, как “put” и "but". Однако, вероятно, одно из них будет отклонено на основе синтаксического анализа, так как глаголы и союзы не играют одинаковую роль в предложении. Аналогично, подлинная синтаксическая неоднозначность имеется в высказывании, типа " He gave her dog biscuits ", где сочетание "her” может функционировать и как прилагательное и как существительное. Но в этом случае неоднозначность может быть решена с помощью ударения и интонации, которые будут сопровождать обе интерпретации.

Предложенные структуры - иерархические, с последовательным потоком информации через цепочку компонентов ИЗ, и неиерархические, без ограничения на поток информации между компонентами.

 Преимущество иерархического подхода в том, что имеется естественный порядок для прикладной программы ИЗ, чтобы вводить речь; синтаксический анализ может осуществляться только на основе лексической информации и т.д. Кроме того, в целом управление системы просто. Однако, имеются много случаев, когда непоследовательные взаимодействия между цепочкой компонентов полезны; например, аспекты просодической, сверхсегментальной структуры высказывания будут релевантны по отношению к фонологической, синтаксической, семантической, и прагматической интерпретации. Непоследовательное взаимодействие может быть достигнуто внутри иерархической модели, передавая все возможные анализы, совместимые с данным компонентом следующему, который затем выбирает подмножество анализов. Но это только тогда сработает, если промежуточные представления, переданные через СПР настолько обогащены, что можно было бы использовать всю проанализированную информацию в следующих компонентах. Таким образом, ввод синтаксического компонента в дополнение к синтаксической информации относительно слов должен включить всю доступную информацию для синтаксического анализа, типа просодической информации, и вся информация, относящаяся семантическому/прагматическому анализу должна быть также включена. Это усложняет схему представления, и дорого в вычислительном отношении, т.к. создает много неправильных гипотез. Неправильных гипотез можно избежать, т.к. информация, в которой отсутствует неоднозначность временно доступна, она закодирована в той части речевого сигнала, который уже проанализирован на более низких уровнях, но в иерархической модели этот способ не применяется, пока ввод не достигает соответствующего компонента в последовательной цепочке.

Неиерархические системы избегают неэффективности, позволяя компонентам применять в наиболее эффективном порядке сложные межкомпонентные связи. Каждый компонент нужно обеспечить средствами, чтобы запрашивать и получить информацию из других компонентов или начинать определенную обработку в другом компоненте. Это требует специальных каналов связи между компонентами в системе. Разработка адекватной системы управления для такой модели невозможна, т.к. должна предусматривать все возможные потоки управления в стадии проекта. Практически, реальные неиерархические модели для СПР были ограничены однородными представлениями из ИЗ и одиночной глобальной структурой данных, как в (blackboard systems) рабочих системах.

Стратегии Обработки

Различные стратегии обработки использовались в разных структурах СПР, чтобы сократить вычисление, требуемое для успешного анализа. И иерархические и неиерархические системы могут работать со способами управления данными как снизу-вверх, так и сверху-вниз при использовании знания, чтобы создать гипотезы относительно ввода. Однако, самые современные СПР используют способ снизу-вверх из-за довольно слабого предсказания речи на основе ИЗ. Аналогично, СПР может исследовать пространство, определяя его глубину и ширину. Большинство систем оперирует с шириной пространства из-за сомнительного или ошибочного характера многих гипотез, но использует подсчитывающие методы, чтобы сохранить размер активного исследуемого пространства. Одна из таких методик, подсчитывающая неудачи, которая включает измерение совокупности множества индивидуальных слов-кандидатов в соотношении с теоретической верхней границей и обработку гипотезы, гарантирует, что СПР найдет наиболее полную подсчитывающую гипотезу для первого высказывания. Однако это не гарантирует, что наиболее привлекательная гипотеза является правильной; эффективность компонентов, которые способствуют порождению гипотез слова, все еще является определяющим фактором в полном представлении системы. Этим оценкам должны отвечать все компоненты, и они должны отражать различные добавления каждого ИЗ. Однако, значение, которое должно быть присоединено к любому ИЗ, должно измениться в соответствии с контекстом. Например, при распознавании безударного и фонетически редуцированного предлога, синтаксический анализ должен чаще обращаться к акустическому анализу, чем при распознавании ударного слога. Кроме того, исследования должны быть оценены с помощью времени. Хотя некоторые схемы оценки, которые использовались в готовых СПР, улучшают эффективность, это связано или по теоретическим причинам, с подсчитывающей методикой, например, подсчитывающей неудачи, или, потому что они были разработаны на основе испытаний и ошибок и оценивались исключительно по эффективности, связанной со временем выполнения, например механизм фокуса внимания в рабочей системе HEARSAY-11.

Анализ речевого сигнала может проходить слева направо через линейный сигнал или из середины островов большей акустической надежности в обоих направлениях. Подход, использующий острова надежности, имеет преимущество в принятии свободных от ошибок фонетических данных за начальную отметку за счет более сложной структуры управления и организации системы, как в HWIM. По-видимому слушатели обращают большее внимание на ударные слоги, которые вообще более ясно произносятся, и следовательно более легко анализируются фонетически. Кроме того, фонологическая структура английского словаря вынуждена быть составленной таким способом, при котором каждое слово может быть получено даже при грубом фонетическом анализе структуры слога вместе с детальным анализом ударного слога. Следовательно, подход, использующий острова надежности по существу правилен, хотя и был бы более эффективен, если обработка началась в ударных слогах.

Текущие Тенденции

Начиная с проекта ARPA в 70-ых имел место период в исследовании речевого понимания, скорее ориентированный на проблемы, чем на построение систем. Многие из этих исследований сосредоточились на акустическо-фонетическом преобразование в результате новых доказательств, показывающих информационное богатство акустического сигнала. Сейчас же возобновлен интерес к построению полных систем, включающий исследования, касающиеся структуры системы. Однако, большинство развивающихся систем, основанных на знаниях, ограничено скорее распознаванием непрерывной речи, чем пониманием. Усовершенствования в акустическо-фонетическом анализе предполагают, чтобы верхние уровни анализа не были определяющими для распознавания непрерывной речи, вопреки преобладающему мнению во времена проекта ARPA. Но проблемы понимания, такие как способы представление знаний, остаются нерешенным.

Системы

Главные СПР, разработанные в проекте ARPA, были HARPY, HWIM, HTEARSAY-11, и SRI/SDC. HARPY оказался наиболее близким по критерию эффективности, определенном для проекта. Однако, структура HARPY требовала составления всего ИЗ в одну конечную сеть, так что язык, воспринимаемый системой был более ограничен, чем в других системах. Система HEARSAY-11 была создана как промышленная система. Несколько СПР были разработаны для Европейских языков, таких как KEAL и MYRTILLE-11 для Французского языка и EVAR для немецкого. Однако, эти системы не превзошли системы ARPA по эффективности или проекту. Так же была создана автоматическая система бронирования места на авиалинии, которая включает непрерывное понимание речи. Эта система, разработанная в Лабораториях Bell, отвечает на телефон, чтобы установить соответствующую бронь. Она использует метод сопоставления целового слова с шаблоном, чтобы распознать слова из словаря, насчитывающего 127 слов.