МЕЖДУНАРОДНЫЙ ЦЕНТР
НАУЧНОЙ И ТЕХНИЧЕСКОЙ
ИНФОРМАЦИИ

ГОСУДАРСТВЕННЫЙ КОМИТЕТ
СОВЕТА МИНИСТРОВ СССР
ПО НАУКЕ И ТЕХНИКЕ

ИНСТИТУТ
ПОВЫШЕНИЯ КВАЛИФИКАЦИИ
ИНФОРМАЦИОННЫХ РАБОТНИКОВ

Б. Р. Певзнер

ИНФОРМАЦИОННО-ПОИСКОВЫЕ СИСТЕМЫ И ИНФОРМАЦИОННО-ПОИСКОВЫЕ ЯЗЫКИ

(Лекции)

Москва
1977

Научный редактор Д.Г. Лахути

ПРЕДИСЛОВИЕ

В этих лекциях даются основные понятия, связанные с информационно-поисковыми системами (ИПС) и информационно-поисковыми языками (ИПЯ).

В структурном отношении лекции делятся на пять глав.

В первой главе только вводятся основные понятия теории ИПС, а раскрытие их содержания дается в последующих главах лекций.

Во второй главе рассматриваются типичные представители документальных систем «с грамматикой» и «без грамматики».

Третья глава посвящена проблеме оценок ИПС.

В четвертой главе описана методика построения ИПС.

В пятой главе приведены основные задачи в области построения и эксплуатации ИПС.

В приложении рассматриваются элементы теории алгоритмов, математической логики и математической лингвистики.

Изложение материала носит в основном компилятивный характер, причем литература, список которой приводится в конце каждой главы, использовалась в разной степени. Иногда из соответствующих источников заимствована только терминология, иногда материал, изложенный в оригинале, был перефразирован в других терминах, и, наконец, иногда имеет место текстуальное совпадение с оригиналом (этот материал дается в тексте без кавычек).

За содержание лекций, их структурное оформление, отбор материала автор несет полную ответственность. Все замечания, полученные автором, будут им тщательным образом изучены.

Глава I. ОСНОВНЫЕ ЭЛЕМЕНТЫ ИПС

Документальная поисковая система, являясь справочным инструментом (аппаратом) поисковой службы, предназначена для отыскания документов, содержание которых соответствует заданному запросу.

Процедуру поиска удобно разделить на два контура. В рамках первого контура решаются две проблемы: семантическая и реализационная. Семантическая проблема включает в себя, во-первых, осмысление введенных в информационно-поисковую систему (алгоритмическую или неалгоритмическую) документов и, во-вторых, осмысление запросов и установление между ними смыслового соответствия. Реализационная проблема включает в себя круг вопросов, связанных с конкретной реализацией обоих семантических процессов. Результатом работы первого контура является выдача адресов (шифров) релевантных (с точки зрения системы) документов.

По этим адресам во втором контуре с помощью различных технических средств (а может быть, и вручную) отыскиваются сами документы. Эти документы или их копии направляются потребителю. Следует иметь в виду, что оба контура могут быть совмещены (например, при поиске в массиве апертурных карт).

В настоящем пособии будут рассматриваться только семантические аспекты построения и функционирования документальных информационно-поисковых систем.

С семантической точки зрения ИПС состоит из трех элементов: информационно-поискового языка, системы перевода (индексирования) на этот язык и логики (см. рис.). Тем самым в ИПС не включаются технические средства ее реализации (такие ИПС принято называть абстрактными).

Рассмотрим более подробно каждый из элементов ИПС.

§ 1. Информационно-поисковый язык

Применение естественного языка (в чистом виде) в ИПС для однозначного описания смыслового содержания документов и запросов связано с весьма значительными трудностями, обуслов-

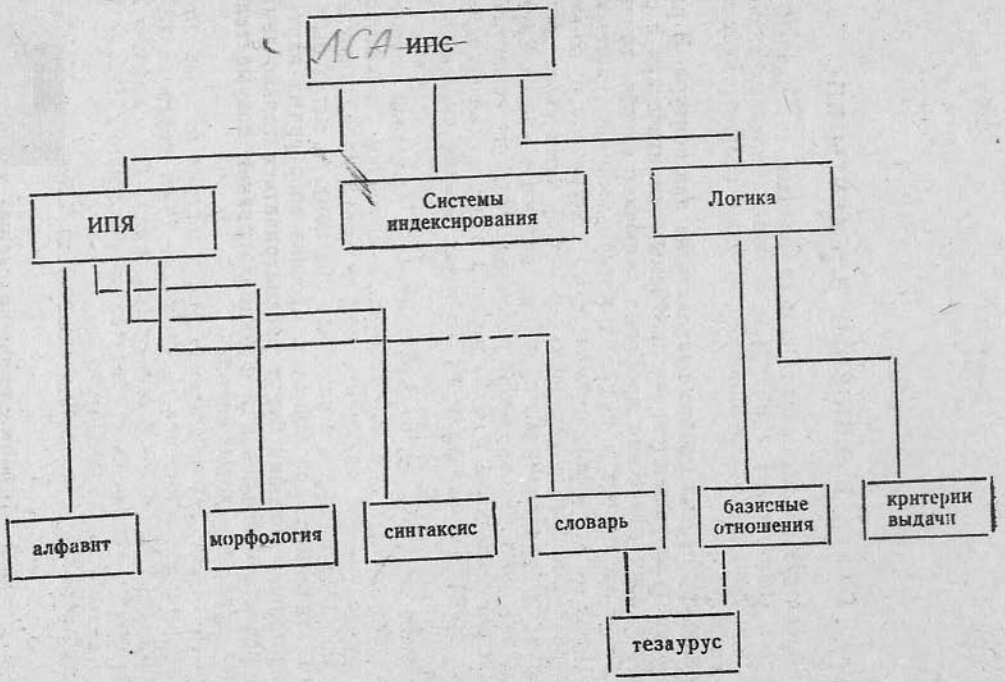


Рис. Элементы ИПС.

ленными, например, наличием синонимов, омонимов. Поэтому вместо естественных языков в ИПС применяются искусственные языки — ИПЯ.

ИПЯ является формальной семантической системой, обеспечивающей передачу (запись) содержания документа в объеме, необходимом для целей поиска. Последнее, в частности, означает, что документ, записанный на этом языке, может быть и не понят человеком.

С другой стороны, сам факт использования слов и выражений естественного языка еще не означает, что в ИПС «используется естественный язык» или «ИПС работает на естественном языке». Во всех нам известных ИПС употребление слов и выражений стандартизировано (в некоторых ИПС оно алгоритмизировано), а также стандартизированы отношения между ними (если они предусмотрены). Поэтому эти системы работают на формальном, а не на естественном языке. Совпадение алфавитов этих языков создает иллюзию того, что в системе используется естественный язык.

Как и всякий естественный язык, ИПЯ состоит из **алфавита** — набора алфавитных символов. Конкретный ИПЯ может использовать как буквенный алфавит, так и цифровой. Выбор алфавита не влияет на семантическую силу языка, а определяется исключительно удобством эксплуатации ИПС.

Из алфавитных символов с помощью **морфологических правил** строятся слова ИПЯ.

В УДК, например, таким правилом является следующее: простые основные индексы (слова ИПЯ системы УДК) строятся как цепочки; при этом после каждых трех цифр, начиная слева направо, следует ставить точку (кстати, точка, двоеточие, знак плюс и другие знаки включаются в алфавит системы УДК). Например, 621.3 — электротехника.

Далее, в некоторых системах при переводе на ИПЯ учитывается не только факт вхождения слова в индексируемый документ, но и те отношения, которые имеют место в данном документе между переводимыми словами. Такие отношения называются **текстуальными** или **контекстуальными.** Системы, учитывающие такие отношения, условно называются системами «с грамматикой».

Следует учесть, что систем «без грамматики», строго говоря, не существует. Под системами «без грамматики» мы понимаем системы, использующие только тривиальные отношения типа: «быть номером текста», «вхождение слова ИПЯ в текст» и т. п., обнаружение которых в исходном тексте не составляет проблемы.

Последним, четвертым элементом ИПЯ является **словарь перевода** с естественного языка на поисковый. В этом словаре каждому слову или осмысленной конструкции естественного языка сопоставляется слово (слова) или конструкция ИПЯ.

Синонимия и омонимия в этих словарях учитывается следующим образом. Близкие по значению слова объединяются в клас-

сы условной эквивалентности (синонимии). Именем этого класса является либо одно слово этого класса, либо дескрипторный номер (номера). Дескриптором называется имя класса слов условной эквивалентности.

Омонимичному слову сопоставляется столько дескрипторов (классов), сколько значений оно имеет в области применения системы.

Например: установка — 302 (вещь);
установка — 303 (действие).

В этих словарях фиксируется основной лексический состав, характерный для области функционирования системы. Такие словари называются положительными. Существует другой способ задания словаря ИПЯ: с помощью отрицательного словаря. Отрицательные словари содержат неинформативные слова (например, предлоги, союзы, наречия и т. д.); при этом способе задания словаря ИПЯ все слова, не включенные в отрицательный словарь, считаются информативными.

Теперь введем классификации ИПЯ* по степени их сложности. Сложность языка является внутренней оценкой системы. Нам не известны такие количественные разумные оценки. Поэтому языки будем оценивать с этой точки зрения качественно.

За основание первой классификации (K_1) принимаются средства, задействованные в ИПЯ. Чем мощнее эти средства, тем сложнее структура выражений этого языка. Согласно предлагаемой классификации, ИПЯ делятся на 4 типа.

К типу K_1^1 относятся ИПЯ, состоящие из одиночных терминов и фиксированных (жестких) словосочетаний. В таких ИПЯ включаются только тривиальные (неизбежные) текстуальные отношения, например вхождение термина в текст.

К типу K_1^2 относятся ИПЯ, в которых текстуальные отношения задаются в виде указателей роли. Указатели роли являются одноместными предикатами (см. приложение). Например «X — продукт химической реакции».

К типу K_1^3 относятся ИПЯ, в которых текстуальные отношения задаются в виде связок. С помощью таких связок на формальном языке фиксируются симметричные n-местные отношения (связи) между словами, вошедшими в поисковый образ документа (запроса). Например, отношение совместного вхождения в некоторый подтекст.

К типу K_1^4 относятся ИПЯ, в которых текстуальные отношения задаются как связками, так и указателями роли. При этом можно выделить два случая.

1. С помощью связок учитывается только факт наличия связи между словами, а с помощью указателей роли — только направление (ориентация) связи. При этом не устанавливается тип (имя) этих связей. Такие текстуальные отношения являются

---

* Рассматриваемые классификации ИПЯ предложены Д. Г. Лахути.

несимметричными недифференцированными n-местными отношениями. Этот тип отношений можно проиллюстрировать с помощью следующих рассуждений.

Пусть в язык системы введен двухместный предикат (связка) $P(x, y)$ — «x и y связаны между собой какой-то связью» и два одноместных предиката (указатели роли) $Q(x)$ — «x главный член конструкции» и $R(x)$ — «x подчиненный член конструкции». Тогда наличие в тексте конструкции, в которой x подчиняет y, можно записать так:

$$P(x, y) \wedge Q(x) \wedge R(y)$$

Отношение синтаксического подчинения является примером такого предиката.

2. С помощью связок или указателей роли выделяются разные типы связей (отношений). Такие отношения являются несимметричными дифференцированными n-местными отношениями.

Приведем пример задания дифференцированных отношений с помощью указателей роли. Пусть в язык введен тот же предикат $P(x, y)$ — «x и y связаны между собой» и несколько разных одноместных предикатов (указатели роли), например субъект (x), т. е. «x — субъект в рассматриваемой конструкции», объект (x), предмет (x), параметр (x). Тогда с помощью этих предикатов конструкции «x воздействует на y» и «x имеет параметр y» можно записать соответственно, как $P(x, y) \wedge$ субъект $(x) \wedge$ объект (y) и $P(x, y) \wedge$ предикат $(x) \wedge$ параметр (y).

Заметим, что на языке с недифференцированными отношениями эти конструкции записались бы одинаково:

$$P(x, y) \wedge Q(x) \wedge R(y).$$

Теперь приведем пример задания дифференцированных отношений с помощью интерпретированных связок и неинтерпретированных указателей роли.

Пусть в ИПЯ введен двухместный предикат, определенный на множестве упорядоченных пар, а именем этого предиката пусть служит пара терминов: «субъект — объект» (или предмет — определение). Он может быть записан в такой форме: субъект — объект (x, y). Он истинен тогда, когда вместо переменной x подставляется термин, являющийся субъектом в анализируемой конструкции, а вместо переменной y — термин, являющийся объектом в этой конструкции. В этом случае и факт связи и интерпретация этого отношения задаются предикатом P.

За основание второй классификации (K_2) принимается сложность контекста, который должен быть проанализирован при индексировании документов. (Нам не известны формальные средства различения сложности контекстов, поэтому сложность контекстов будем определять интуитивно.)

6737

Согласно этой классификации ИПЯ делятся на 6 типов.

К типу $К_2^1$ относятся ИПЯ, при переводе на которые используется только пословная замена слов индексируемого документа дескрипторами.

К типу $К_2^2$ относятся ИПЯ, система индексирования на которые включает не только пословный перевод, но и систему простого анализа и опознавания в документе фиксированных в словаре словосочетаний.

К типу $К_2^3$ относятся ИПЯ, при переводе на которые используется система индексирования языка $К_2$ в комплексе с системой алгоритмического распознавания омонимии. Объектом анализа второй системы является включающий омоним локальный контекст, размер которого не превышает одного предложения (при этом не проводится синтаксический анализ этого контекста).

К типу $К_2^4$ относятся ИПЯ, система индексирования на которые включает в себя в качестве основного элемента систему синтаксического (может быть, упрощенного) анализа предложения. Система устанавливает недифференцированные связи (например, синтаксическое подчинение). Размер контекста равен одному предложению.

К типу $К_2^5$ относятся ИПЯ, система индексирования на которые включает систему синтаксического анализа текста, устанавливающую дифференцированные *n*-местные связи; причем в процессе анализа предложения система имеет возможность обращаться к предыдущим или последующим предложениям и использовать полученную из них информацию для анализа данного предложения.

К типу $К_2^6$ относятся ИПЯ, при переводе на которые используется анализ целых смысловых фрагментов документа, а может быть, и всего документа.

Такая система семантико-синтаксического анализа не только устанавливает связи между смысловыми конструкциями, но и определяет как роль, в которой выступают слова в этих конструкциях, так и тип устанавливаемых связей.

Тем самым система устанавливает несимметричные дифференцированные отношения (связи). Результатом такого индексирования может быть либо принадлежащая ИПЯ (например, ИПЯ системы с грамматикой) конструкция, имеющая сложную структуру, либо простой индекс в системе типа УДК.

В заключение рассмотрения предложенных классификаций ($К_1$ и $К_2$) следует еще раз подчеркнуть, что $К_1$ упорядочивает ИПЯ по степени сложности структур их слов (выражений), а $К_2$ упорядочивает их по степени сложности контекстов естественного языка, анализируемых при переводе (индексировании) на соответствующий ИПЯ.

§ 2. Система индексирования

Процедура перевода с естественного языка на ИПЯ называется индексированием. Результат такого перевода документа на ИПЯ называется поисковым образом документа (ПОД). Результат перевода запроса называется поисковым образом запроса (ПОЗ). Процедура индексирования используется при вводе документов в фонд. Она при неалгоритмической реализации весьма трудоемка (использует в большом объеме интеллектуальный труд документалиста-индексатора) и, кроме того, дорогостояща. Помимо этого неалгоритмический характер индексирования ограничивает семантические возможности даже потенциально мощных ИПЯ. Поэтому вопрос о системах индексирования будет в дальнейшем рассматриваться под углом зрения возможности автоматизации (алгоритмизации) этого процесса.

Рассмотрим различные типы систем индексирования.

К первому типу относятся системы свободного индексирования. При этом способе из индексируемого документа индексатор вписывает в ПОД слова (или словосочетания), которые, по его мнению, отражают содержание документа. Кроме этого, элементами ПОДа могут быть слова, отсутствующие в этих документах. Выписанные элементы упорядочиваются в алфавитном порядке. Такой упорядоченный набор слов представляет собой ПОД. Описанный процесс является принципиально неалгоритмическим процессом.

При втором методе, который условно назовем методом полусвободного индексирования, индексатор выписывает из документа слова и словосочетания, так же как и при свободном индексировании. Однако выписанные элементы сравниваются затем с фиксированным словарем, не найденные в нем устраняются, а оставшиеся, упорядоченные в алфавитном порядке, представляют собой ПОД.

Третий способ индексирования основан на статистическом подходе, а именно: выбор слов (выражений) исходного текста, подлежащих включению в ПОД, производится на основе статистической обработки текста, при котором его слова рассматриваются как знаки, не имеющие семантических значений.

Предлагались различные статистические критерии для выбора слов из текста в ПОД. Перечислим некоторые из них:

$$|F-R|>k;\ \frac{F}{F+R}<k;\ \frac{F}{R}>k,$$

где F — относительная частота употребления слова в документе;
R — относительная частота употребления слова в представительном массиве документов.

Возможность алгоритмизации индексирования статистического типа не вызывает сомнений.

Легко видеть, что в основе всех этих формул лежит общая идея, согласно которой информационная значимость слова опре-

деляется расхождением частоты его употребления в данном документе и во всем потоке рассматриваемых документов. В качестве примера рассмотрим два принципа определения упомянутого выше расхождения.

Согласно первому вычисляется расхождение между частотой употребления слова в потоке документов данной тематики (монотематический поток) и частотой встречаемости этого же слова в многотемном потоке документов (политематический поток).

Второй принцип основан на вычислении расхождения частоты употребления слова в потоке текстов данной тематики и частоты этого же слова в потоке текстов тематики, далекой от данной («противоположной» тематики).

Следует особо подчеркнуть, что самостоятельного практического применения статистические методы еще не нашли, они носят в основном вспомогательный характер как при построении ИПС, так и при ее эксплуатации.

К четвертому типу относятся системы индексирования, контролируемые заданным словарем. В некоторых системах этот словарь используется как помощник специалисту-индексатору (например, в системах УДК). В других системах такой словарь является элементом алгоритмов индексирования. Алгоритм сводится к следующему: каждое слово текста сравнивается с точностью до основы со словарем. Слово, одновременно встретившееся в тексте и в словаре, записывается в ПОД. Чаще всего в ПОД записывается не само слово текста, а соответствующий ему дескриптор.

§ 3. Логика системы

Третьим элементом ИПС является логика системы. Она может состоять из двух элементов — базисных (парадигматических) отношений между словами ИПЯ и критерия выдачи*.

Роль базисных отношений (если они зафиксированы) сводится к следующему. Дело в том, что в естественном языке имеет место такой факт: одни и те же события или явления могут описываться в разных терминах. Значит, может оказаться, что в запросе и в релевантном ему документе могут быть употреблены разные слова. Более того, на практике приходится отыскивать релевантные документы, в которых речь идет о более частных понятиях, чем в запросе. Для того чтобы не потерять все такие документы, в ИПС вводятся базисные отношения: отношения между словами — дескрипторами ИПЯ. Они в отличие от текстуальных отношений не зависят от контекста. Поэтому их можно (как это и делается в некоторых системах) задавать списком. Базисные отношения, увеличивая семантическую силу системы, позволяют сформулировать запрос в терминах, отличных от терминов, употребленных в релевантных документах.

---

* Критерий выдачи является необходимым элементом логики, а базисные отношения могут отсутствовать.

Фиксированные базисные отношения могут быть заданы различными способами. Например, с помощью структуры слов ИПЯ, как это сделано в УДК, с помощью системы ссылок, с помощью деревьев (графов).

Следует иметь в виду, что введение какого-то базисного отношения в систему, строго говоря, является противоречивым актом. С одной стороны, эта мера способствует уменьшению потерь релевантных документов, с другой — может привести к увеличению шума.

Для компенсации отсутствия базисных отношений в системах, где они не зафиксированы, применяются две различные стратегии индексирования, носящие неалгоритмический характер. Первая стратегия сводится к вписыванию в ПОД слов, не встретившихся в самом документе («избыточное индексирование»). Это в основном синонимы и более широкие термины по отношению к тем, которые употреблены в индексируемом документе.

Вторая стратегия связана с переформулировкой запроса, представлением его в виде серии подзапросов, уточняющих и раскрывающих его основное содержание. По каждому из этих подзапросов проводятся поиски, объединенные результаты которых служат ответом на основной запрос.

Теперь рассмотрим второй (необходимый) элемент логики — критерий выдачи. Этот элемент в информатике называется также критерием смыслового соответствия. Критерий выдачи позволяет формально решать вопрос о выдаче или невыдаче того или иного документа, он реализуется в алгоритме поиска. Критерии выдачи могут быть как строго фиксированными, так и меняться от поиска к поиску. Рассмотрим некоторые типичные фиксированные критерии выдачи.

Критерий «на вхождение». Документ формально выдается тогда и только тогда, когда его ПОД содержит все дескрипторы запроса. Другими словами, документ формально выдается, если множество ($M_з$) дескрипторов запроса входит в множество ($M_д$) дескрипторов этого документа ($M_д \supset M_з$).

Критерий на вхождение с учетом базисных отношений. Документ выдается в том случае, если для каждого дескриптора запроса в его ПОДе встретился либо сам этот дескриптор, либо дескриптор, связанный с дескриптором запроса базисным отношением.

Критерий на вхождение с учетом текстуальных и базисных отношений. Этот критерий в основном совпадает с предыдущим; различие заключается в том, что сравнение дескрипторов запроса и документа должно осуществляться с точностью до совпадения текстуальных отношений, в которые их прообразы вступают соответственно в запросе и документе.

Существуют *критерии* выдачи, *основанные на учете весовых коэффициентов.* Их суть сводится к следующему. Каждому информативному слову в запросе приписывается весовой коэффициент (W_i), причем, чем большее значение придает потребитель како-

му-то слову, тем больший весовой коэффициент он ему приписывает. Сумма всех весовых коэффициентов в запросе должна быть константой — ($\Sigma W_i = \text{const}$). Выдача на эти запросы эшелонируется в зависимости от суммы весовых коэффициентов слов запроса, совпавших* со словами, употребляемыми в документе. Количество эшелонов выдачи, а также соответствующие каждому из них значения суммы весовых коэффициентов (порог), определяются разработчиком системы в процессе ее отладки.

В заключение рассмотрения элементов ИПС введем термин, широко используемый в информатике, а именно — тезаурус. Тезаурус — это словарь поискового языка, в котором зафиксированы различные типы логических (семантических) отношений. Другими словами, тезаурус — это множество слов, на котором определены некоторые семантические отношения (например, синонимия, отношение общего к частному, часть — целое, причина — следствие и др.).

Итак, поисковую систему характеризуют языком, системой индексирования и логикой. При этом можно считать, что, например, любые два словаря определяют две разные поисковые системы. Такая точка зрения не всегда представляется естественной. Так, изменение состава словаря даже на один дескриптор (например, в процессе отладки системы), строго говоря, приводит к необходимости рассмотрения новой поисковой системы. Аналогично дело обстоит с локальной корректировкой системы базисных отношений**. Однако такие две системы — новую и старую — хотелось бы рассматривать как два варианта одной и той же системы. С этой точки зрения поисковую систему хотелось бы определить не конкретным набором дескрипторов, базисных отношений, конкретной системой индексирования и конкретным критерием выдачи, а, скорее, типом допустимых выражений, отношений между ними, их соответствия выражениям естественного языка. Этим, в частности, определяется желание ввести классификацию поисковых языков (и систем индексирования на них) по степени их сложности (см. стр. 7).

§ 4. Функционирование ИПС

В любой ИПС реализуются два семантических процесса: индексирование и поиск***. Процесс индексирования был обсужден выше. Перейдем к рассмотрению поиска, определяемого критерием выдачи.

---

* Точность совпадения может быть разной: *а)* совпадение с точностью до окончаний или только основ; *б)* совпадение с точностью до базисных отношений; *в)* совпадение с точностью до базисных и текстуальных отношений.

** Больше того, в такой ситуации систему без фиксированного критерия выдачи (он может меняться в зависимости от стратегии поиска, задаваемой потребителем) не приходится рассматривать как одну конкретную систему.

*** См. стр. 12. Индексирование реализует осмысление вводимых в ИПС документов и запросов, а поиск проверяет наличие смыслового соответствия между ними.

В зависимости от способа организации информационного массива различают два типа поиска: прямой и обратный (инверсный).

Прямой поиск используется в тематически неупорядоченных массивах. В этих массивах документы рассматриваются в порядке возрастания их регистрационных номеров (адресов). Элементом поискового массива являются поисковые образы документов.

Перечислим основные операции прямого поиска: *1.* Индексирование документов; *2.* Индексирование запросов; *3.* Сравнение каждого ПОДа с ПОЗом согласно выбранному критерию выдачи.

Таким образом, при реализации прямого поиска вопрос о выдаче документов решается для каждой пары ПОД — ПОЗ. Число таких пар при поиске по одному запросу равно числу документов в информационном массиве. А это естественным образом приводит к тому, что время поиска существенно зависит от размера информационного фонда. Примером реализации прямого поиска могут служить системы перфокарт с поиском на сортировках.

При инверсном поиске элементом поискового массива является так называемое инверсное (ассоциированное) множество. Количество инверсных множеств равно числу слов ИПЯ (в частности, дескрипторов в ИПС «без грамматики»). Каждому слову ИПЯ сопоставляется свое инверсное множество, именем которого является это слово. Элементами любого инверсного множества являются номера документов, поисковые образы которых содержат либо дескриптор, совпадающий с его именем, либо дескриптор, связанный с этим именем базисным отношением, если они зафиксированы в системе.

Перечислим основные операции инверсного поиска: *1.* Индексирование документов; *2.* Индексирование запросов; *3.* Образование массива инверсных множеств для каждого дескриптора (для некоторых они, естественно, могут быть пустыми); *4.* Выбор из этого массива инверсных множеств, имена которых совпадают с дескрипторами запроса. Массив инверсных множеств может быть упорядочен, например, по возрастанию дескрипторных номеров; *5.* Проведение теоретико-множественных операций, определяемых логикой запроса, над выбранными инверсными множествами.

Время инверсного поиска разделяется на два интервала: на первом (t_1) — создается массив инверсных множеств, на втором (t_2) — над этими множествами проводятся логические операции.

При прямом поиске не образуется массива инверсных множеств, поэтому не затрачивается время t_1. В этом отношении прямой поиск выгоднее. Однако «чистое» время поиска t_2 при инверсном поиске значительно меньше, чем при прямом. Оно определяется в основном не размером информационного массива, а числом дескрипторов в запросе (длиной запроса) и средней длиной инверсного множества. Если при прямом поиске параллельно

обрабатывается не один запрос, а целая серия, то эта разница становится меньше.

ЛИТЕРАТУРА

к главе I

Михайлов А. И., Черный А. И., Гиляревский Р. С. Основы информатики. Гл. 6. М., «Наука», 1968.

Теория и практика научно-технической информации. Сборник лекций. М., ВИНИТИ, 1969, с. 264—276. 285—290.

Глава II. ОПИСАНИЕ РАЗЛИЧНЫХ ТИПОВ СИСТЕМ

В этом разделе будут описаны представители систем, использующих ИПЯ различных типов. Следует иметь в виду, что, как правило, конкретный ИПЯ, строго говоря, нельзя отнести к какому-то одному типу, поэтому деление на типы следует рассматривать, скорее, как относительное, чем как абсолютное.

§ 1. Система «Унитерм»

Рассмотрение начнем с ИПС, использующих языки типа $К_1^1$ и $К_2^2$. Типичным представителем таких систем является система «Унитерм». Описание начнем с примера. Пусть в информационный центр, использующий систему «Унитерм», поступил реферат: «Производственные информационные системы. Всесторонне рассматриваются проблемы информационного обслуживания на предприятиях для целей руководства, а также описываются схемы производственных информационных систем управления (ИСУ), этапы их создания, вопросы применения в них ЭВМ».

Из этого реферата, например, методом свободного индексирования, выбираются ключевые слова, затем они упорядочиваются в алфавитном порядке.

ПОД: информационная система, информационное обслуживание, ИСУ, предприятие, применение, проблема, производственный, руководство, создание, ЭВМ, этап.

Проанализируем формальную выдачу этого документа на различные запросы, используя критерий выдачи «на вхождение».

Запрос 1. Информационное обслуживание на предприятиях. $ПОЗ_1$: информационное обслуживание, предприятие. На этот запрос рассматриваемый документ будет формально выдан в качестве релевантного документа, так как все слова ПОЗа нашлись в ПОДе.

Запрос 2. Разработка ИСУ. $ПОЗ_2$: ИСУ, разработка. Рассматриваемый документ, являясь релевантным запросу 2, не будет формально выдан на него. Для устранения этой потери в процессе индексирования этого документа в его ПОД следовало бы вписать слово «разработка», являющееся синонимом слова «создание».

Запрос 3. Применение вычислительной техники в сфере информационного обслуживания. $ПОЗ_3$: вычислительная техника, информационное обслуживание, применение. Документ будет выдан при условии, что в процессе его индексирования в ПОД было вписано словосочетание «вычислительная техника», являющееся более общим понятием по отношению к понятию «ЭВМ».

Анализ запросов 2 и 3 показал: если в системе не заданы базисные отношения, то для уменьшения потерь приходится вписывать в ПОД не только те слова, которые встретились в документе, но и синонимы и более общие понятия*.

Запрос 4. Руководство предприятиями. $ПОЗ_4$: предприятие, руководство. На этот запрос рассматриваемый документ будет формально выдан, однако он не релевантен этому запросу. Причиной выдачи является случайная комбинация ключевых слов в ПОДе, являющаяся следствием игнорирования текстуальных отношений между ключевыми словами при индексировании. Одним из способов учета таких отношений является введение дополнительных словосочетаний.

Запрос 5. Создание производственных ЭВМ. $ПОЗ_5$: производственный, создание, ЭВМ. На этот запрос рассматриваемый текст будет формально выдан, однако он не релевантен этому запросу. Если бы в процессе индексирования документа в его ПОД было бы вписано словосочетание «производственные информационные системы» (а не ключевое слово «производственный») и словосочетание «информационная система», то ошибочной выдачи не было бы. Однако такое индексирование приведет к потере этого текста на запрос об информационных системах. Таким образом, введение словосочетания в ПОД, с одной стороны, может привести к уменьшению шума, а с другой — к увеличению потерь, поэтому при решении вопроса о введении словосочетаний в словарь принимаются компромиссные решения.

Теперь опишем непосредственно систему «Унитерм», предложенную Таубе. Ее предметной областью является химия и химическая технология.

ИПЯ «Унитерм». Язык системы относится к следующим типам: $К_1^1$, $К_2^2$. Алфавит языка состоит из 26 латинских букв. Морфологические правила образования слов ИПЯ совпадают с аналогичными правилами английского языка. Синтаксические средства, т. е. текстуальные отношения, отсутствуют. В первом варианте системы словарь отсутствовал. Система индексирования относится к системам свободного индексирования.

Логика системы. В системе отсутствуют базисные отношения. Для их компенсации используются стратегии, описанные на стр. 12. Критерий выдачи относится к типу «на вхождение».

---

* Либо использовать стратегию, связанную с переформулировкой запроса (см. стр. 12).

§ 2. Система «Пусто-Непусто-2»

К языкам типов $К_1^1$, $К_2^3$ относится ИПЯ системы «Пусто-Непусто-2» (ПНП-2). Областью ее функционирования является электротехника. Она предназначена для поиска и обработки вторичных документов (рефератов, библиографических описаний, аннотаций), записанных на русском и английском языках.

ИПЯ системы ПНП-2. Алфавит языка состоит из 10 арабских цифр (0, 1, 2, . . ., 9). Морфологическими правилами построения слов ИПЯ (дескрипторов) являются правила образования десятичных чисел из цифр.

Текстуальные отношения в системе отсутствуют. Основным элементом ИПЯ является русско-дескрипторный и англо-дескрипторный словарь, в который включены одиночные слова естественных языков и, как исключения, словосочетания. Большинству элементов словаря соответствует один дескриптор, словам омонимичного характера сопоставлено столько дескрипторов, сколько семантических значений они имеют в электротехнике.

Система индексирования является, вообще говоря, системой пословного перевода с русского и английского языка на язык системы. Она реализована на ЭВМ. Логика системы содержит фиксированные базисные отношения и критерий выдачи «на вхождение» с учетом базисных отношений.

В системе предусмотрены три типа отношений: синонимия ($Р_0$), прямое подчинение ($Р_1$), обратное подчинение ($Р_2$).

Эвристические приемы введения этих отношений сводятся к следующему:

1. Замена в ПОДе или ПОЗе дескриптора *А* на дескриптор *В* не должна привести к изменению результатов поиска. Дескрипторы *А, В,* связываются отношением $Р_0$.

2. Если в ПОЗе содержится дескриптор *А,* а в ПОДе — дескриптор *В* и все остальные дескрипторы запроса входят в этот ПОД, и если документ должен выдаваться на запрос в первом эшелоне выдачи (эшелон «Да»), то дескриптор *А* и *В* связываются отношением $Р_1$. Если в ПОЗе содержится *В,* а в ПОДе *А,* то при прочих равных условиях документ не выдается на этот запрос.

3. Если в ПОЗе содержится *А,* а в ПОДе — *В* и остальные дескрипторы запроса входят в этот ПОД, и если документ должен выдаваться на запрос во втором эшелоне выдачи «может быть» («МБ»), то *А* и *В* связываются отношением $Р_2$. Если же в ПОЗе встретился *В,* а в ПОДе — *А,* то документ не выдается на этот запрос.

4. Если в ПОЗе встретился *А,* в ПОДе — *В* и все остальные дескрипторы ПОЗа входят в ПОД (документ № 1 должен выдаваться в эшелоне «Да»), и если, с другой стороны, в ПОЗе встретился дескриптор *В,* а в ПОДе — *А* и все остальные дескрипторы ПОЗа входят в ПОД (документ № 2 должен выдаваться в эше-

лоне «может быть»), то дескрипторы A и B связываются отношением P_1 (A, B) и отношением P_2 (B, A).

Критерий выдачи формулируется в терминах пустоты и непустоты двух множеств (M_1 и M_2):

M_1 — множество дескрипторов запроса, не сравнимых (т. е. не совпадающих и не связанных никакими базисными отношениями) ни с какими дескрипторами документа;

M_2 — множество дескрипторов запроса, которые не совпадают и не связаны отношениями P_1 и P_0, а связаны только отношениями P_2 с какими-то дескрипторами документа. В зависимости от распределения пустоты и непустоты множеств M_1 и M_2 сравниваемый текст выдается (в эшелоне «Да» или «МБ») или не выдается на запрос.

Критерий выдачи удобно описать в виде следующей таблицы.

M_1	M_2	
пусто	пусто	текст выдается в эш. «Да»
пусто	непусто	текст выдается в эш. «МБ»
непусто	пусто	—*
непусто	непусто	текст не выдается

* Из определения множеств M_1 и M_2 видно, что такая ситуация невозможна.

§ 3. Система «Кристалл»

Теперь рассмотрим систему «Кристалл», язык которой относится к типам ИПЯ $К_1^2$ $К_2^5$. Вначале кратко опишем функционирование этой системы. На ее вход поступают вторичные документы, написанные на естественном языке. Индексатор составляет квазиреферат каждого вторичного документа. Процесс составления сводится к преобразованию каждой фразы вторичного документа в назывное предложение. Из такого квазиреферата с использованием словаря неинформативных слов (отрицательного словаря) выбираются значимые слова.

ИПЯ располагает четырьмя указателями роли: логическое подлежащее (ЛП), предподлежащее (ПП), факторное подлежащее (ФП), теневое подлежащее (ТП). Логическим подлежащим могут быть термины, раскрывающие основную тему документа или его фрагмента. Предподлежащее соответствует терминам, характеризующим свойства и процессы, методы и состояния. Факторным подлежащим могут быть термины, характеризующие действующие агенты, условия, предикаты и регистраторы действия. Теневое подлежащее — это категория вспомогательных терминов, использующихся для уточнения любой из трех перечисленных выше категорий подлежащих и раскрывающих их качественные и количественные характеристики.

Каждое подлежащее обозначается соответствующими двузначными числами. Логическому подлежащему соответствует одно из трех чисел:

20 — объект исследования (название материи, вещества, энергии, абстрактных объектов);

60 — название инструментов, оборудования, приборов;

30 — название составных частей объектов, относящихся к 20 и 60.

Предподлежащему сопоставляется одно число:

10 — предподлежащее (название процессов, операций, состояний, свойств).

Факторному подлежащему сопоставляются два числа:

50 — название действующих факторов, агентов;

70 — название среды, в которой протекает процесс (время, место, условия).

Для обозначения теневых подлежащих используются следующие значения младшего разряда двузначных чисел:

1 — качественная характеристика основного подлежащего;

2 — количественная характеристика основного подлежащего;

3 — единица измерения;

4 — начало интервальной количественной характеристики;

5 — конец интервальной количественной характеристики;

6 — вакансия;

7 — вакансия.

Машинная запись представляет собой коды неравномерной длины, состоящие из словоформ естественного языка. Этим словоформам приписываются указатели роли в виде цифрового кода. Существует специальный алгоритм перекодирования (автокодирования) ключевых слов в пятисимвольный код.

Перечислим основные алгоритмические процедуры: *1.* Опознать и отсечь окончание (имеется словарь окончаний); *2.* Опознать и отсечь суффикс (имеется словарь суффиксов); *3.* Остаток слова преобразовать в пятисимвольный код.

Информационный массив системы делится на несколько тематических подмассивов. В легкой промышленности их восемь. Например: общие вопросы текстильного производства, экономика, организация производства, ткачество. Каждый такой подмассив снабжается номером, который приписывается входному документу или его фрагменту, тематика которого соответствует данному подмассиву. Документы внутри подмассива расставляются в порядке возрастания их инвентарных номеров.

Индексирование запроса сводится к выбору ключевых слов, приписыванию им соответствующих указателей роли и весовых коэффициентов. Кроме ключевых слов индексатор вписывает в ПОЗ синонимы и родовые и/или видовые термины, снабжая каждый из них указателем роли и весовым коэффициентом. Критерий выдачи относится к типу критериев, основанных на весовых коэффициентах (см. стр. 12).

В зависимости от суммарного веса терминов, употребленных

в документах, последние выдаются в разных эшелонах. Число эшелонов равно 3.

Обозначим сумму весов всех терминов запроса S_Σ. В первом эшелоне выдаются документы, в которых употреблены все термины запроса, т. е. сумма весовых коэффициентов равна S_Σ. Во втором эшелоне выдаются документы, в которых встретились такие (не все) термины запроса, сумма весовых коэффициентов которых лежит в диапазоне $\frac{S_\Sigma}{2} \leqslant S < S_\Sigma$. В третьем эшелоне должны выдаваться все оставшиеся в фонде документы. Однако эта выдача ограничивается заданием в формулировке поискового предписания (запроса) некоторого «критического» веса (порога).

Документы попадают в третий эшелон тогда, когда они не попали в первые два эшелона и когда в них встретились термины запроса, сумма коэффициентов которых равна или больше «критического» веса.

В службе АСИОР, семантической базой которой является «Кристалл», хорошо разработана обратная связь с потребителем.

§ 4. Система «Синтол»

Система «Синтол», по замыслу авторов, может работать в различных режимах: как без грамматики, так и с грамматикой (простой или развитой). Поэтому ИПЯ системы представляет собой семейство информационных языков, обладающих различной семантической силой. Языки, входящие в это семейство, разработаны таким образом, что язык с большей семантической силой включает в себя целиком (в качестве подъязыка) языки с меньшей семантической силой. Согласно нашим классификациям языки системы «Синтол» относятся к следующим типам: $К_1^1$, $К_1^2$, $К_1^3$ и $К_4^1$, (см. стр. 12), а также к $К_2^3$, $К_2^4$ и $К_2^5$.

Слова, входящие в словарь системы, распределены по небольшому числу «категорий»; синтаксис языка формулируется в терминах этих «категорий». Кратко опишем структуру словаря. Она может быть представлена в виде дихотомического дерева, имеющего 3 уровня:

Уровень *а*.

Предикаты служат для унификации выражений естественного языка. Например: выражение: «отравляющее воздействие углекислого газа на нервную систему» может быть записано на ИПЯ с использованием предиката: «действие *х*, отравляющее, нервная система». Вместо *х* может быть подставлен определенный элемент словаря (например, углекислый газ). Таким предикатам сопоставляются множества слов-аргументов (они названы элементами) по правилу, которое задается *а priori*.

Уровень *в*.

Элементы делятся на «предметы» и «функции». Предметы — это существа, тела, объекты; функции — это слова, обозначающие свойства «предметов».

Уровень *с*.

Функции (свойства предметов) делятся на два типа: пассивные свойства предметов («состояния»), например «перерождение», и динамические свойства («действия»). В словаре системы каждому слову естественного языка приписан один из признаков: «предмет», «состояние», «действие».

Теперь опишем систему синтагмических (текстуальных) отношений. Она задается в виде древовидной структуры:

Уровень *а*.

Формальные отношения (координатные отношения) устанавливаются между близкими по значению словами. Реальные отношения — это отношения типа часть — целое.

Уровень *в*.

Реальные отношения делятся на динамические (консекутивные) и статические. Первые устанавливаются в том случае, когда один элемент влияет на характер другого или на его состояние.

Уровень *с*.

Статические отношения, в свою очередь, делятся на ассоциативные (отношения субъекта к действию, отношение действия к субъекту и т. д.) и предикативные отношения. Предикативные отношения устанавливаются между такой парой элементов, один из которых является «предикатом». Два слова, связанные одним из отношений, называются синтагмой, а сами отношения — синтагматическими.

Для каждого типа отношений задаются правила ориентации синтагм. Так, для ассоциативного отношения между термином, выражающим «действие», и любым другим термином ориентация зависит исключительно от той роли, которую выполняет второй гермин по отношению к «действию»: от логического субъекта или агента «действия» к самому «действию», от «действия» к его дополнениям. С помощью синтаксического анализа и правил ориентации синтагм в тексте могут быть опознаны синтагмы, определен тип отношений между ними и установлено направление между элементами опознанной синтагмы.

При установлении ассоциативных отношений для уточнения записи смысла на поисковом языке используются операторы — инструментальный, локализации, цели, признак.

В рассмотренном языке применяются правила «развертывания» («вычисления») синтагм, которые (синтагмы) явно не заданы в тексте. Возьмем последовательность двух синтагм *(а, в)*, *(в, с)* и отношения между словами R_1 *(а, в)* и R_2 *(в, с)*. С помощью правил «развертывания» может быть образована новая синтагма *(ас)* с новым отношением R_3.

Эти правила задаются в виде таблиц. Правила «свертывания» синтагм (обратные «развертыванию») называются модуляциями.

В язык системы предполагается вводить средства, позволяющие выделить во фразе основной элемент (тему), сопутствующие элементы (парамегры: время, место) и точку зрения автора (жанр: историческая справка, популяризация, эксперимент и т. д.). Предполагается, что тема, время, место и жанр могут быть выделены с помощью ответа на вопросы соответственно: что? когда? где? как?

Об автоматизации алгоритма индексирования в литературе имеются довольно скудные сведения. Сообщается о том, что проведено опытное автоматическое индексирование на ЭВМ, массив заиндексированных документов составлял 500 текстов. Сам алгоритм, к сожалению, не описан. Из этих сведений многие вопросы остаются неясными, например, проводилось ли индексирование на том же масиве, на базе которого составлялся алгоритм, либо массив был новым. Не приводится никаких сведений о качестве индексирования документов.

Логика системы «Синтол». Базисные (парадигматические) отношения задаются в виде многоуровневой структуры. На первом уровне лексика грубо делится на «поля» (физиология, психология, социология и т. д.). «Поля», в свою очередь, делятся на «части» (2-й уровень). «Части» — это довольно общие названия разделов, включенных в определенное «поле» (для физиологии: «системы», «аппараты», «жидкости»). На третьем уровне «части» делятся на «главы» — широкие семантические классы, которые могут быть использованы как ключевые слова («кровеносная система», «пищеварительный аппарат»). «Главы» делятся на «секции», которые включают ключевые слова более узкие, чем ключевые слова, принадлежащие «главам». И наконец, секции делят-

ся на «подразделения», состоящие из очень специализированных ключевых слов.

Поиск. Предусматривается возможность преобразования запроса в логическую форму с использованием функций — дизъюнкции (или), конъюнкции (и) и отрицания (не) (см. приложение).

Сам поиск можно проводить на разных уровнях: на тематическом уровне, на уровне слов (ИПЯ дескрипторного типа без грамматики) и на уровне синтагм (ИПЯ дескрипторного типа с грамматикой). На синтагматическом уровне предусматривается поиск с учетом неинтерпретированных связей (устанавливается факт наличия связи, но не устанавливается ее имя), с учетом интерпретированных связей, которым присваиваются соответствующие имена, но без операторов и, наконец, с учетом интерпретированных связей и с операторами.

Естественно, что каждый тип поиска, соответствующий разным критериям выдачи, обеспечивает определенные параметры (не указываемые авторами), причем точность поиска, как правило, возрастает по мере усложнения языка.

§ 5. Система «Смарт»

Система «Смарт» используется в основном как экспериментальный инструмент для оценки эффективности различных семантических средств, вводимых в ИПС. Поэтому система включает в себя различные типы ИПЯ.

Языки системы. В системе предусмотрены различные типы словарей.

1. Отрицательный словарь содержит термины, использование которых запрещается при составлении поискового образа документа.

2. Словарь тезаурусного типа. Так, авторы системы называют словарь, учитывающий классы условной эквивалентности. Сформулируем основные принципы построения такого словаря:

1) очень редко встречаемые термины не должны быть включены в этот словарь;

2) очень общие термины тоже должны быть исключены из словаря;

3) особое внимание следует уделять незначимым словам, прежде чем их исключить из словаря. Так, слово *hand* (рука) должно быть включено в словарь по биологии. Если же это слово имеет высокую частоту встречаемости за счет выражения «on the other hand» (с другой стороны), то оно опускается при составлении словаря;

4) омонимичным терминам приписывают столько дескрипторных номеров, сколько они имеют значений в данной области функционирования системы;

5) в каждый класс эквивалентности должны включаться синонимичные термины, частоты употребления которых в среднем равны друг другу.

В системе «Смарт» использовано 700 классов синонимии.

Рассматриваются два типа словарей: словарь, элементами которого являются полные английские слова, и словарь, состоящий из корней слов. На выборке в 50 000 слов было получено два словаря корней: полный словарь, содержащий 2800 различных корней, и частичный словарь, содержащий 900 корней, частота употребления которых в этой выборке была не ниже 7.

Работая со словарями корней при индексировании, приходится проводить операции, связанные с выделением корней из слов, встретившихся в индексируемом документе. Для этого в систему вводится словарь суффиксов и окончаний (ed, ing, s, fy и т. д.).

В словаре суффиков каждому суффиксу сопоставлены синтаксические коды, с помощью которых обозначаются части речи, для образования которых используется данный суффикс. Все это помогает снимать при анализе текстов омонимию частей речи.

Кроме рассмотренных выше различных типов словарей предусматриваются также словари фраз, т. е. словосочетаний. Различают два вида таких словарей: словарь статистических словосочетаний и словарь синтаксических словосочетаний. Первый тип словарей основан на совместной встречаемости различных классов эквивалентности. Для опознавания этих фраз в тексте используются две стратегии.

Алгоритм опознает «статистическое словосочетание», если, и только если все его компоненты встретились в анализируемом тексте или в одном из его предложений. Этот алгоритм не учитывает наличия синтаксических отношений между компонентами словосочетаний. Словарь синтаксических фраз содержит не только их лексические компоненты, которые должны быть опознаны в тексте, но и информацию о возможной синтаксической сочетаемости между компонентами.

С помощью этого словаря в тексте опознаются компоненты словосочетания и определяется наличие или отсутствие между ними синтаксических отношений. Словосочетания, как правило, состоят из 2-х и 3-х компонентов (как исключение, их число достигает 6). Синтаксический анализ описан авторами очень мало, не приводится никаких результатов этого анализа, за исключением отдельных, специально подобранных примеров. Согласно нашим классификациям ИПЯ «Смарт» относится к следующим типам: $К_1^1$, $К_1^2$, $К_1^3$ (статистические словосочетания), $К_1^4$ (см. стр. 12; синтаксические словосочетания), $К_2^1$, $К_2^2$, $К_2^3$, $К_2^4$.

Логика системы «Смарт». Между классами эквивалентности могут задаваться базисные отношения. Критерий выдачи является аналитической функцией косинуса угла между двумя векторами: вектором документа и вектором запроса. Эта функция выражается следующим образом:

$$\cos(q, d) = \frac{\Sigma q_i \cdot d_i}{(\Sigma (d_i)^2 \cdot \Sigma (q_i)^2)^{1/2}},$$

где q и d суть n-мерные векторы в пространстве дескрипторов, представляющие запрос q и документ d, а q_i и d_i — соответственно их i-е координаты, принимающие значения 1 или 0 в зависимости от того, входит ли i-ый дескриптор в соответствующий дескрипторный образ или нет.

Вместо 1 можно использовать другой весовой коэффициент. Распределение весов по дескрипторам может основываться, например, на частоте их встречаемости или на учете словарных характеристик: омониму присваивается меньший вес, чем не омониму. Эта функция принимает значения в интервале от 0 до 1.

Строго говоря, в соответствии с заданным запросом система так ранжирует весь массив документов, что в первый эшелон (ранг) попадает документ, для которого вычисленное значение косинуса максимально. Остальные документы располагаются по степени убывания соответствующего им значения функции. На практике не все документы массива, конечно, выдаются (ранжируются). Формальная выдача ограничивается некоторым порогом.

ЛИТЕРАТУРА

к главе II

Белоногов Г. Г., Богатырев В. И. Автоматизированные информационные системы. Гл. 10, 17, 15. М., «Сов. радио», 1973.

Дейк ван М., Сплит ван Ж. Информационная служба в условиях информационного взрыва. М., ВИНИТИ, 1972. 80 с.

Добронравов И. С., Лахути Д. Г., Малинин С. Г. Вопросы разработки семантической поисковой системы с грамматикой для электротехники. — В кн.: Проблемы построения и развития алгоритмических документальных ИПС. Отделение ВНИИЭМ. М., 1973, с. 48—55.

Сборник переводов по вопросам информационной теории и практики, № 10. СИНТОЛ. М., ВИНИТИ, 1968, с. 38—47, 50—52, 68—72, 76—80.

Сэлтон Г. Автоматическая обработка, хранение и поиск информации. М., «Сов. радио», 1973, с. 36—38, 288—292.

Теория и практика научно-технической информации. Сборник лекций. М., ВИНИТИ, 1969, с. 297—298, 223—227, 230—232.

Хисамутдинов В. Р., Легоньков В. И., Авраменко В. С., Тарасов В. И. Автоматизированная система информационного обеспечения разработок «АСИОР». Препринт № 39. М., ИПМ, 1970, с. 10—18.

Юпатов Е. П., Коровякова И. Д., Тарасов В. И. Отраслевая автоматизированная система информационного обеспечения «Кристалл-Легпром». М., ЦНИИТЭИЛегпром, 1970.

Глава III. ОЦЕНКИ ПОИСКОВЫХ СИСТЕМ

На стр. 4 мы дали определение (абстрактной) поисковой системы как системы, состоящей из ИПЯ, системы индексирования и логики. Однако всякая ИПС всегда реально существует в виде некоторого поискового массива документов, записанных на определенном языке (языках), поступающих из определенного источника (источников), некоторого коллектива людей и оборудования, реализующих правила индексирования и поиска, наконец, некоторого круга потребителей, служащих источником зап-

росов. Все это вместе мы будем называть поисковой службой. В этой главе будет идти речь об оценках поисковых систем, а не служб. В связи с этим кратко будут рассмотрены три вопроса:

A — вопрос об определении оценки;

B — вопрос об эталоне содержательной выдачи;

C — вопрос о представительности массивов документов и запросов.

Вопрос *A*.

Все мыслимые оценки поисковых систем можно было бы разделить на два класса, которые уместно называть внешними и внутренними оценками; их можно было бы назвать также функциональными и нефункциональными оценками. Внешние, или функциональные, оценки основаны на сравнении результатов работы системы с результатами идеального содержательного поиска, осуществляемого экспертом. Они предполагают понятие релевантности. Внутренние оценки могли бы основываться на таких структурных качествах системы, как сложность, степень близости к человеческой логике или естественному языку, степень алгоритмичности и т. п. Насколько нам известно, до сих пор не предложено сколько-нибудь разработанных конкретных внутренних оценок, поэтому они здесь не будут рассматриваться.

Итак, рассматриваем только внешние оценки. Под оценкой (способом, методом оценки) понимается более или менее алгоритмическая процедура, которая любому оцениваемому объекту (из данной области) ставит в соответствие некоторый другой объект, называемый значением оценки. Полностью алгоритмическую процедуру оценки мы будем называть формальной оценкой. Различают два типа оценок:

— оценки-описания, значения которых характеризуют непосредственно систему безотносительно к другим системам;

— оценки-шкалы, значения которых определяют сравнительные достоинства различных поисковых систем.

От «оценки-описания» требуется, чтобы ее значения позволяли достаточно полно судить о существенных свойствах оцениваемых объектов, например предсказывать их поведение в тех или иных конкретных условиях. В этом случае «оценка-описание» называется эффективной. От «оценки-шкалы» требуется, чтобы ее значения упорядочивали множество оцениваемых объектов (например, различных ИПС), не вступая при этом в противоречие с существующими у нас содержательными представлениями о сравнительных достоинствах этих объектов (и в этом случае мы называем «оценку-шкалу» здравой). Содержательные представления о сравнительных достоинствах систем мы будем называть содержательной оценкой. Таким образом, здравая формальная оценка не должна противоречить содержательной. Следует иметь в виду, что одна и та же формальная оценка может рассматриваться и использоваться как «оценка-шкала», так и «оценка-описание».

Всякая оценка ориентируется на те или иные свойства (например, семантические) оцениваемых систем. Если доброкачественность формальной оценки («оценки-шкалы» или «оценки-описания») считать совпадающей с ее здравостью, то эта доброкачественность существенно зависит от содержательной точки зрения на сравнительные достоинства оцениваемых поисковых систем. Эта точка зрения определяется той задачей, для которой мы хотим использовать ИПС, и теми условиями, в которых мы ее используем. Например, требования, предъявляемые к ИПС, осуществляющей правовой патентный поиск, отличаются от требований к отраслевым ИПС. Следовательно, с изменением условий и решаемой задачи может меняться точка зрения, а с ней и используемая формальная оценка. Поэтому не может существовать такой универсальной оценки, область здравости которой включала бы все осмысленные точки зрения.

Вопрос *В*.

Определение полноты системы связано с определением содержательной выдачи на каждый запрос. Существует несколько способов определения этой выдачи:

— сплошной просмотр всего экспериментального массива. Достоинством этого способа является надежность, недостатком — трудоемкость. Трудоемкость этого метода примерно оценивается такими данными: 10 человеко-часов на 1 запрос при поиске по массиву в 4000 документов. Однако при применении этого метода приходится считаться с тем, что при самом аккуратном проведении содержательного поиска по одному и тому же запросу разными экспертами или даже одним и тем же экспертом в разное время получается разная содержательная выдача. Однако обусловленная неоднозначностью содержательного эталона степень колебаний ($\approx 5\%$) значений наиболее употребительных формальных оценок поисковой системы, основанных на полноте и точности, существенно меньше, чем степень колебания (50%) самой содержательной выдачи;

— метод документа-источника («метод Клевердона») состоит в том, что по некоторым документам массива, выбранным более или менее случайно, составляются запросы с таким расчетом, чтобы каждый документ-источник был релевантен составленному по нему запросу.

За значение полноты принимается доля запросов, по которым система выдала документ-источник в общем числе запросов. Этот метод привлекает своей минимальной трудоемкостью. Однако применимость этого метода основана на предположении, что вероятность нахождения системой документа-источника по одному запросу равна вероятности нахождения по этому запросу произвольного релевантного документа в данном массиве. Но это не так. В экспериментах группы А. В. Соколова искажение значения полноты при использовании этого метода достигало 25%;

— метод контрольных документов сводится к следующему. По запросу, полученному по произвольно выбранному документу-

источнику, проводится содержательный поиск путем сплошного просмотра массива (начиная, например, с документа-источника) до нахождения первого релевантного документа, который объявляется контрольным. Значение полноты для системы считается теперь как доля запросов, по которым система выдала контрольный документ в общем количестве запросов. Значение полноты, полученное по этому методу, мало отличается от значения полноты, вычисленной путем сплошного просмотра. Считается, что на массивах порядка 5—10 тыс. документов этот метод может сократить сплошной просмотр по крайней мере в 3—4 раза;

— метод объединения формальных выдач применяется при сравнении нескольких поисковых систем («оценка-шкала»). Он состоит в том, что по каждому запросу эксперт просматривает только те документы, которые выдавались хотя бы одной из этих поисковых систем. Содержательной выдачей считается совокупность обнаруженных релевантных документов, и относительно нее определяется полнота (которая отличается от истинной полноты каждой из рассматриваемых систем).

Вопрос *С.*

Вопрос о представительности массива документов и массива запросов, выбранных для определения формальных оценок, является еще далеко не изученным. Считается, что более или менее устойчивые оценки (колебания не превышают ±5%) можно получить на массиве в 4000 документов, массив запросов при этом должен быть порядка нескольких сотен (100—200).

В заключение раздела приведем формулы для вычисления наиболее употребительных оценок полноты (потерь) и точности (шума). Полноту и точность соответственно обозначим R и P.

$$R_i = \frac{a_i}{c_i}, \qquad P_i = \frac{a_i}{b_i},$$

где a_i — число релевантных документов, формально выданных системой на i-запрос;

b_i — число всех формально выданных на i-запрос системой документов;

c_i — число всех релевантных документов, соответствующих i-запросу.

Среднее значение полноты и точности определяется так:

$$R = \frac{\sum^{N} \frac{a_i}{c_i}}{N}, \qquad P = \frac{\sum^{N} \frac{a_i}{b_i}}{N} \text{ (средняя относительная оценка)}$$

$$R = \frac{\sum^{N} a_i}{\sum^{N} c_i}, \qquad P = \frac{\sum^{N} a_i}{\sum^{N} b_i} \text{ (суммарная относительная оценка),}$$

где N — число поисков.

Величины $(1-P)$ и $(1-R)$ называются соответственно шумом и потерями.

В связи с оценкой системы «Смарт» Сэлтон ввел нормированную полноту (R_N) и нормированную точность (P_N)

$$R_N = \frac{1}{N} \sum_{i=1}^{N} \frac{n_i}{n},$$

где N — число документов в массиве;

n — число всех релевантных документов в массиве;

n_i — число релевантных документов, выданных до i-го ранга включительно.

$$P_N = \frac{1}{N} \sum_{i=1}^{N} \frac{n_i}{i},$$

где i — номер ранга.

ЛИТЕРАТУРА

к главе III

Лахути Д. Г. и др. О проблеме оценки поисковых систем. — «НТИ», сер. 2, 1970, № 1, с. 24—34.

Михайлов А. И., Черный А. И., Гиляревский Р. С. Основы информатики. М., «Наука», 1968, с. 302—316.

Певзнер Б. Р. Сравнительная оценка русского и английского варианта системы «Пусто-Непусто-2». — «НТИ», сер. 2, 1972, № 6.

Семантические проблемы информатики. М., ВИНИТИ, 1971, с. 6—34.

Соколов А. В. Исследование потерь информации и информационного шума в дескрипторных информационно-поисковых системах. — «НТИ», 1965, № 12, с. 23—28.

Глава IV. МЕТОДИКА ПОСТРОЕНИЯ ИПС

В этой главе рассматриваются этапы построения и отладки ИПС.

Этап I. Формулировка назначения системы. В зависимости от задачи, для которой предназначается строящаяся поисковая система, к ней могут предъявляться различные требования, например:

— система должна обеспечивать поиск на патентную новизну;

— система должна обеспечивать поиск на патентную чистоту;

— система должна обеспечивать обычный информационный поиск.

Этап II. Выбор класса системы. Если система должна быть ориентирована на узкоспециализированные запросы в выбранной области, то предпочтение, как правило, отдается дескрипторным системам.

Этап III. Выбор языка системы. Рекомендуется на первой стадии выбирать простые языки, например языки «без грамматики», а затем, в процессе отладки и, может быть, экспериментальной или даже технологической эксплуатации усложнять этот язык, если это потребуется. Целесообразность перехода к сложным языкам может быть выявлена в процессе анализа причин потерь и шума.

Этап IV. Разработка дескрипторного словаря.

1. Работа начинается с составления словника, при этом могут использоваться различные методы:

— частотный метод;

— метод, основанный на свободном индексировании опытного массива текстов по выбранной тематике;

— метод, основанный на выборе терминологии из существующих классификаторов, предметных указателей, различных справочников и других аналогичных источников.

Как показывает опыт, основная терминологическая часть словника, его ядро, составляется относительно просто и быстро. В дальнейшем словник должен корректироваться в процессе отладки системы.

Далее слова, вошедшие в словник, объединяются в классы условной эквивалентности, каждому классу сопоставляется дескрипторный номер. В один класс могут объединяться синонимы и полусинонимы (данной области). Словосочетанием объявляется набор слов в том случае, если эти слова связаны в системе базисных отношений с разными элементами и если этот набор слов в качестве самостоятельной терминологической единицы должен быть связан с каким-то другим элементом языка. Словосочетанию приписывается один дескрипторный номер.

Омонимом объявляется слово, имеющее несколько значений в выбранной области; неразличение значений этого слова (что эквивалентно невыделению его в качестве омонима) может увеличить шум системы при поиске по запросам, в которых употреблено это слово. Омонимичным словам приписывается столько дескрипторных номеров, сколько значений они имеют. Сложным (например, двухкорневым) словам могут приписываться несколько (например, два) дескрипторных номеров.

При разработке словаря полезно иметь в виду эмпирическую зависимость между объемом словаря и числом документов, которые должны быть заиндексированы (методом свободного индексирования) в процессе построения словаря: $V=39\sqrt[4]{T}$, где V — объем словаря, T — число заглавий, по которым проводится индексация; $V=118\sqrt[4]{Д}$, где $Д$ — число рефератов (а не заглавий), по которым проводится индексация.

2. Разработка системы базисных отношений.

Различают несколько типов связей (отношений):

— общее — частное, или род — вид (машина — двигатель);

— существительное — образованное от него прилагательное (вентиль — вентильный);

— причина — следствие («скольжение (щеток)» — «искрение»);

— эмпирические связи.

Эмпирические связи не имеют общелогической или общеязыковой наглядности первых трех типов связей, но их способность уменьшить потери определяется довольно тонкими особенностями языка и системы понятий обслуживаемой предметной области. Примером может служить связь дескрипторов «переменный ток» и «асинхронный», полезность которой основана на том, что эти дескрипторы часто встречаются в сочетании с дескриптором «двигатель», а асинхронные двигатели являются двигателями переменного тока. Вообще говоря, подобные сведения могут быть почерпнуты, например, из учебников по электротехнике или из опыта специалистов. Но дело в том, что «эмпирические» связи могут быть в большей степени, чем другие типы связей, противоречивыми, так как, предотвращая потерю нужных документов, они всегда приводят к выдаче какого-то числа нерелевантных документов и общий эффект введения такой связи может быть оценен только после того, как эта связь (или ее отсутствие) проявит себя в каких-то фактически приведенных поисках.

Эксперименты, проведенные группой Д. Г. Лахути, дают основания считать, что полнота выдачи за счет наличия базисных отношений возрастает примерно в 2 раза для эшелона «да», а для всей выдачи — в три раза по сравнению с соответствующим «унитермовым» вариантом системы.

Этап V. Выбор типа индексирования. В зависимости от степени автоматизации проектируемой системы выбирается тип системы индексирования (см. стр. 10).

Этап VI. Выбор критерия выдачи. Доброкачественность выбранного критерия (см. стр. 12) определяется в процессе отладки системы, да и то лишь косвенным путем — с помощью внешних оценок.

Этапы III—V проводятся параллельно.

Все описанные выше 6 этапов могут быть реализованы следующим образом.

Методом свободного индексирования обрабатываются 500 (1 — 500) документов, составляется фрагмент тезауруса. В полученном поисковом массиве проводятся 30—50 содержательных и формальных поисков (Методы определения содержательной выдачи, см. стр. 27). Вычисляются следующие значения полноты и точности (R_{500}, P_{500}). Анализируются причины потерь и шума (см. стр. 16) и вносятся корректировки в тезаурус.

Рекомендуется отлаживать систему на искусственных запросах, составляемых сериями с таким расчетом, чтобы они максимально широко охватывали дескриптор поискового языка. Кроме того, отлаживать систему можно на запросах, полученных путем случайного выбора фраз из имеющегося массива текстов. И наконец, систему можно отлаживать сразу на реальных запросах, од-

нако этот метод имеет принципиальный недостаток, так как адаптирует систему к достаточно узкому кругу интересов потребителей.

Затем все операции повторяются на новом массиве (501—1000) документов. Разница лишь в том, что на этом этапе уже используется выбранная система индексирования. Вычисленные величины P_{1000} и R_{1000}, как правило, больше P_{500} и R_{500}. Система на этом этапе также корректируется.

Все описанные операции повторяются до тех пор (примерно до накопления массива порядка 4000 документов), пока точность и полнота, полученные на отлаженном массиве, не окажутся близки к параметрам, полученным на новом массиве.

ЛИТЕРАТУРА

к главе IV

Лахути Д. Г. Вопросы отладки и оценки дескрипторных поисковых систем. — В кн.: Семантические проблемы информатики. М., ВИНИТИ, 1972, с. 6—34.

Певзнер Б. Р., Лахути Д. Г. Методика построения иноязычных систем дескрипторного типа «без грамматики». — В кн.: Проблемы построения и развития алгоритмических документальных ИПС. М., Отделение ВНИИЭМ, 1973, с. 34—42.

Черный А. И. Общая методика построения тезауруса. — «НТИ», сер. 2, 1968, № 5.

Глава V. СОВРЕМЕННОЕ СОСТОЯНИЕ ИПС И ОСНОВНЫЕ ЗАДАЧИ В ОБЛАСТИ ИХ ПОСТРОЕНИЯ

Современное состояние отечественных ИПС представлено в виде таблицы, предложенной Ю. И. Шемакиным*.

Мы кратко остановимся на вопросах, которые ждут своего решения и над которыми работают создатели ИПС.

Для удобства изложения эти вопросы разделим на два типа: теоретические и эксплуатационные (технологические). Рассмотрение начнем с теоретических вопросов.

1. Выявление семантической силы различных элементов ИПС, т. е. определение влияния этих элементов на параметры системы, например на точность и полноту.

2. Построение и анализ систем с «грамматикой», обеспечивающих довольно высокую полноту при удовлетворительной точности поиска. При этом надо решать целый комплекс задач, основные из них — это автоматическое индексирование, связанное, в частности, с алгоритмическим синтаксическим анализом, а также с выявлением текстуальной синонимии и учетом ее при поиске.

3. Анализ алгоритмических иноязычных систем индексирования на ИПЯ и исследование возможности одновременного построения отраслевых ИПС для разных естественных языков.

---

* Добавлены данные о системе «АСИНИТ».

Таблица

Системы	Характеристики																	
	Тип			Режим		Тип ИПЯ			Критерий выдачи						Автоматизация			
	документальный	фактографический	логический	ИРИ	ретроспективный	классификация	дескрипторный	библиографич. ссылки	логические функции	аналитическ. функции	весовые функции	полное совпадение	полное вхождение	частичное вхождение	поиска	индексирования	классификации документов	управления обратной связью
Пусто-Непусто-2	+	—	—	+	+	—	+	—	+	—	—	—	+	+	+	+	—	—
Ордината	+	—	—	+	+	—	+	—	+	—	—	—	—	+	+	—	—	—
Реферат	+	—	—	+	+	—	+	—	—	—	—	—	+	+	+	—	—	—
БИТ	+	—	—	+	—	—	+	—	+	—	—	—	—	+	+	—	—	—
Кристалл	+	—	—	+	+	—	+	—		—	+	—	—	+	+	—	—	—
Сетка-5	+	—	—	+	+	—	+	—	+	—	—	—	—	+	+	—	—	—
ДИПСИ	+	—	—	—	+	—	+	—	+	—	—	—	—	+	+	—	—	—
ИПС-Фтор	+	—	—	+	+	—	+	—	+	—	—	—	—	+	+	—	—	—
Аргон	+	—	—	—	+	—	+	—	+	—	—	—	—	+	+	—	—	—
Компас	+	—	—	+	+	—	+	—	+	—	—	—	—	+	+	—	—	—
СИАЛ	+	+	—	—	+	—	+	—	+	—	—	—	—	+	+	—	—	—
ИПС ЦБТИ	+	—	—	+	—	+	+	—	+	—	—	—	—	+	+	—	—	+
Кузбасс	+	—	—	—	—	—	+	—	+	—	—	—	—	+	+	—	—	—
Спектр	+	—	—	—	+	—	+	—		—	—	—	—	+	+	—	—	—
ИПС по конструк. матер.	+	—	—	—	—	—	+	—	+	—	—	—	—	+	+	—	—	—
АСИНИТ	+	+	—	+	+	—	+	—	—	—	+	—	—	—	+	—	—	—

4. Автоматизация процедур, необходимых для построения различных элементов автоматизированных ИПС: автоматический отбор словаря, автоматический выбор оптимального критерия выдачи, автоматическое построение системы базисных отношений.

5. Построение информационно-логических систем, моделирующих процесс вывода новых фактов из старых. (Наиболее разработана эта проблема для химии.)

6. Проблема оценок поисковых систем. Методы определения (вычисления) релевантности и пертинентности и методы, позволяющие по заданному потребителем запросу до поиска предсказать наиболее вероятное значение полноты и точности при поиске по этому запросу. Последнее связано, в частности, с классификацией запросов по определенным группам и определением для каждой из них гарантированных значений оценки.

7. Определение методов расчета экономической эффективности работающей поисковой службы.

Трудность решения всех теоретических вопросов связана с тщательными и, к сожалению, весьма трудоемкими проверками формулируемых гипотез.

Эксплуатационные вопросы охватывают широкий круг проблем, связанных с поддержанием параметров работающей поисковой службы в заданных пределах, с анализом работы цепи обратной связи «потребитель — система» и с многими другими чисто технологическими и организационными вопросами функционирования поисковой службы.

ЛИТЕРАТУРА

к главе V

Шемакин Ю. И. Современное состояние разработок и использование в СССР автоматизированных и механизированных ИПС. М., ВИНИТИ, 1972. (Госкомитет по науке и технике).

ПРИЛОЖЕНИЕ

ЭЛЕМЕНТЫ ТЕОРИИ АЛГОРИТМОВ, ЭЛЕМЕНТЫ МАТЕМАТИЧЕСКОЙ ЛОГИКИ, ЭЛЕМЕНТЫ СТРУКТУРНОЙ ЛИНГВИСТИКИ

Элементы теории алгоритмов

Под алгоритмом для некоторого класса задач понимается некоторое общее правило, метод, с помощью которого решение любой задачи этого класса может быть найдено чисто механически, если это решение существует. Так, может быть определен алгоритм на интуитивном уровне. Однако, как показывает опыт, все известные алгоритмы могут быть «промоделированы» в том или ином смысле. А. Черч высказал предположение, называемое тезисом Черча, согласно которому можно отождествить интуитивное понятие алгоритма с одним из точных его определений (например, машина Тьюринга, нормальный алгоритм Маркова и т. д.). Это значит, что для любого алгоритма в интуитивном понимании можно построить алгоритм в смысле некоторого точного определения, эквивалентный ему и дающий в применении к одинаковым исходным данным всегда одинаковые результаты. Можно доказать эквивалентность различных точных определений алгоритма, но нельзя, в строгом смысле, доказать тезис Черча, поскольку речь идет о связи точного понятия алгоритма с неточным интуитивным понятием, не поддающимся строгой математической экспликации. Однако весь опыт работы с алгоритмами любого рода является подтверждением тезиса Черча.

Перед тем, как излагать точные алгоритмы, дадим общую схему алгоритма.

Алгоритм $И$ оперирует с конкретными объектами. Он задается конечным предписанием B, входным алфавитом E, выходным алфавитом C, рабочим алфавитом A; другими словами, задается четверкой символов $<B, E, C, A>$. (Под алфавитом понимается конечное непустое множество символов; конечные последовательности символов из некоторого алфавита назовем словами над этим алфавитом) Рабочий алфавит A состоит из символов, которые могут быть использованы для записи промежуточных результатов

работы алгоритма. Предписание B должно быть составлено так, чтобы: во-первых, определяемые им операции были во всех деталях однозначно осуществимы; во-вторых, порядок их выполнения был однозначно определен; в-третьих, исполнение предписания было воспроизводимо. Теперь опишем один из алгоритмов в точном смысле, а именно машину Тьюринга.

Имеется бесконечная лента, разделенная на клетки. В каждой клетке может быть записан один символ входного алфавита $S_1, S_2, \ldots, S_m$ *; имеется рабочий орган, который может «видеть» один символ в клетке, заменять его на другой, может двигаться влево, вправо или стоять на месте. В каждый момент времени рабочий орган может находиться в одном из состояний конечного множества $\{q_i\}$. Набор состояний составляет внутренний алфавит $q_0, q_1, q_2, \ldots, q_n$. В этом алфавите выделяется два состояния: q_1 — начальное и q_0 — конечное. Работой рабочего органа управляется программа: неупорядоченный набор команд определенного вида. Каждая команда состоит из двух частей: левой и правой. В левой части записывается исходная ситуация (пара символов): символ входного или рабочего алфавита и состояние рабочего органа (символ внутреннего алфавита). В правой части записывается новая ситуация (пара символов): вообще говоря, другой символ внешнего алфавита и другое состояние рабочего органа. Кроме того, в каждой команде указывается направление движения рабочего органа Θ (R — вправо, L — влево, C — стоять на месте). Вид команды: $S_n q_l / S_m q_k \Theta$, в частности, может быть $n=m$ и/или $l=k$.

Команда читается так: рабочий орган, находясь в состоянии q_l, «видит» на ленте символ S_n, заменяет этот символ на S_m, переходит в состояние q_k и сдвигается в Θ.

Правила применения программы:

1) определяется очередная ситуация (т. е. состояние рабочего органа и символ на ленте);

2) в списке команд отыскивается команда, левая часть которой совпадает с этой ситуацией;

3) выполняются действия, определяемые ситуацией, записанной в правой части этой команды;

4) см. п. 1.

Одна и та же команда может применяться несколько раз. К началу работы на ленту подается начальная информация (начальное слово), изображаемая в символах входного алфавита. Работа машины складывается из следующих один за другим тактов, осуществляемых одной из команд программы. При этом происходит преобразование начальной информации в промежуточную информацию, которая может изображаться в символах как рабочего, так и входного алфавитов.

---

* Результат работы машины (выходное слово), как правило, изображается в символах входного алфавита, т. е. C совпадает с E.

В качестве начальной информации на ленту можно подать любую конечную последовательность знаков (слово) входного алфавита. Если после конечного числа тактов машина останавливается, подавая сигнал об остановке (переходя в состояние q_0) и при этом на ленте оказывается изображенной некоторая конечная (результирующая) информация (выходное слово), то говорят, что машина применима к начальной информации. Если же после конечного числа тактов остановка не наступает и сигнал об остановке никогда не подается (машина не переходит в состояние q_0), то говорят, что машина не применима к начальной информации.

Для примера приведем программу удвоения каждого символа исходного слова.

Внешний (входной) алфавит: a, b, λ^*+

Алфавит состояний: q_a, q_b, q_{1a}, q_{1b}, q_1, q_2, q_3, q_0, q_i

Рабочий алфавит: b', a'

Так выглядит лента к началу работы алгоритма:

a	$в$	a	$в$	$в$	$+$				

В начале работы рабочий орган, находясь в состоянии q_1, «видит» символ a, запоминает его, в эту клетку записывает a' и движется вправо, переходит символ «+», размножает «a» два раза, затем возвращается за другим символом (первым слева нештрихованным). Аналогичным способом удваиваются все символы исходного слова. Картина на ленте окончательно выглядит так:

a'	$в'$	a'	$в'$	$в'$	$+$	a	a	$в$	$в$	a	a	$в$	$в$	$в$	$в$

Далее рабочий орган стирает исходное слово. Теперь составим программу машины Тьюринга для решения этой задачи.

$aq_1 \mid a'q_aR$
$bq_1 \mid b'q_bR$

эти команды «запоминают» размножаемый символ.

$bq_a \mid bq_aR$
$aq_a \mid aq_aR$
$bq_b \mid bq_bR$
$aq_b \mid aq_bR$
$+q_a \mid +q_aR$
$+q_b \mid +q_bR$

команды сдвига вправо, к свободному месту для записи «удвоенного» символа.

$\lambda q_a \mid aq_{1a}R$
$\lambda q_{1a} \mid aq_2L$
$\lambda q_b \mid bq_{1b}R$
$\lambda q_{1b} \mid bq_2L$

эти команды находят свободное место, записывают на него удвоенный символ и поворачивают назад рабочий орган.

---

* λ — символ пустой клетки на ленте.

$aq_2 \mid aq_2 L$ $bq_2 \mid bq_2 L$ $+\ q_2 \mid +q_2 L$	команды возврата к очередному, еще не размноженному символу.
$a' q_2 \mid a' q_1 R$ $b' q_2 \mid b' q_1 R$	эти команды замыкают цикл переноса и удвоения очередного символа.
$+\ q_1 \mid +\ q_3 L$ $b' q_3 \mid \lambda q_3 L$ $a' q_3 \mid \lambda q_3 L$ $\lambda q_3 \mid \lambda q_0 C$	программа стирания исходного слова.

В программе не может быть двух команд, имеющих одну и ту же левую часть и разные правые (однозначность).

Существует другое точное определение интуитивных алгоритмов — нормальные алгоритмы А. А. Маркова.

Алгоритм служит для преобразования заданного (начального) слова R_1 в конечное S, которое называют результатом работы алгоритма. В процессе преобразования алгоритм строит конечное число промежуточных слов ($R_2, \ldots, R_n$). Каждое слово представляет собой конечную последовательность символов алфавита.

Запись этих алгоритмов состоит из команд типа подстановок. Каждая команда состоит из двух частей P и Q:

P — символ или конечная последовательность символов (то, вместо чего производится подстановка в начальное или промежуточное слово);

Q — символ или конечная последовательность символов (то, что подставляется в начальное или промежуточное слово).

Если R может быть преобразовано в S с помощью однократной подстановки, то R и S называют смежными словами. Цепочка слов $R_1, R_2, R_3, \ldots, R_n$ называется дедуктивной, если для всех i от 1 до n R_i и R_{i+1} суть смежные слова.

Нормальный алгоритм представляет собой упорядоченный перечень команд в отличие от программы машины Тьюринга. Команды, как правило, располагаются в столбец. Команда имеет вид: $R \to Q$. Команда останова обозначается так: $R \to \cdot S$. Алгоритм работает следующим образом: в программе (список команд) отыскивается первая команда сверху, которая может быть применена к самому левому вхождению* символа в начальное или промежуточное слово, и осуществляется подстановка в это слово, задаваемая этой командой.

Для примера рассмотрим программу, которая удваивает каждый символ исходного слова **ababb**. (Рядом с каждой командой будем записывать слово, переработанное с ее помощью.)

---

* Если слово L является частью слова M, то говорят о вхождении слова L в слово M. Так, например, в слове *babaabba* имеются два вхождения слова *ab:* одно из них, начиная со второй буквы, считается самым левым.

1. $*a \rightarrow a'a'$	$a'a'babb*$
2. $*b \rightarrow b'b'$	применяется для слов, начинающихся с „b“
3. $a'a \rightarrow a'a'a'$	
4. $a'b \rightarrow a'b'b'$	$a'a'b'b'abb*$
5. $b'b \rightarrow b'b'b'$	
6. $b'a \rightarrow b'a'a'$	$a'a'b'b'a'a'bb*$
Работает команда 4	$a'a'b'b'a'a'b'b'bb*$
Работает команда 5	$a'a'b'b'a'a'b'b'b'b'*$
$* \rightarrow \lambda$	$a'a'b'b'a'a'b'b'b'b'$
$b \rightarrow b''$ (эта команда в данном случае работает 6 раз подряд)	$a'a'b''b''a'a'b''b''b''b''$
$a' \rightarrow a$ (команда в данном случае работает 4 раза подряд)	$aab''b''aab''b''b''b''$
$b'' \rightarrow b$ (эта команда в данном случае работает 6 раз подряд)	$aabbaabbbb$
$a\lambda \rightarrow .a$	
$a\lambda \rightarrow .b$	

Элементы математической логики

Алгебра логики

Объектами, с которыми оперирует математическая логика, являются высказывания. Высказывания — это такие утверждения (предложения), по отношению к которым можно ставить вопрос, истинны ли они или ложны. Например: снег белый (истинное высказывание); $2 \times 2 = 5$ (ложное высказывание). Из простых высказываний могут быть построены сложные, при этом используются логические операции типа: «или» (дизъюнкция), «и» (конъюнкция), «если . . .,то» (импликация), «не» (отрицание).

Введем знаки, которыми обозначаются эти операции, или логические связки.

Дизъюнкция — $\vee$

Конъюнкция — ., $\wedge$, &

Импликация — $\rightarrow$, $\supset$

Отрицание — —, $\neg$

Приведем примеры сложных высказываний: «Снег белый» или «$2 \times 2 = 5$» (дизъюнкция); «6 делится на 3» и «6 делится на 2» (конъюнкция); «Если число делится на 2, то оно четное» (импликация); «Неверно, что миллион является самым большим числом в натуральном ряду» (отрицание).

Сложные высказывания, построенные с помощью этих операций, также являются либо истинными, либо ложными. Существенно при этом то, что их истинность или ложность определяется (вычисляется) только истинностью или ложностью высказываний, из которых они построены и которые могут быть, в свою очередь, как простыми, так и сложными. Иначе говоря, вышеперечисленные операции являются логическими функциями. Область их определения состоит из двух элементов: «истина» (и) и «ложь» (л), называемых истинностными значениями. Область значений логических функций также состоит из этих двух элементов. Эти функции задаются таблицами истинности.

Дизъюнкция $X \vee Y$

X	Y	$X \vee Y$
и	и	и
и	л	и
л	и	и
л	л	л

Конъюнкция $X \wedge Y$

X	Y	$X \wedge Y$
и	и	и
и	л	л
л	и	л
л	л	л

Импликация $X \supset Y$

X	Y	$X \; Y$
и	и	и
и	л	л
л	и	и
л	л	и

Отрицание $\neg X$

X	$\neg X$
и	л
л	и

Две логические функции называются эквивалентными, если при одном и том же распределении истинностных значений аргументов они имеют одинаковое значение. Эквивалентность обозначается так: $\equiv$.
Например:

$$(a \vee b) \wedge (c \vee d) \equiv b \vee a) \wedge (d \vee c).$$

Раздел математической логики, занимающийся функциями описанного типа, называется логикой высказываний, или алгеброй логики. Особое место в алгебре логики занимают логические функции, которые истинны при любых значениях аргументов и которые всегда называются истинными высказываниями, или законами алгебры логики.

Перечислим основные законы алгебры логики.

1. Закон исключенного третьего: $a \vee \neg a$.
2. Закон противоречия: $\neg (a \wedge \neg a)$.
3. Коммутативный закон: $a \vee b \equiv b \vee a$, $a \wedge b \equiv b \wedge a$.
4. Закон ассоциативности:

$$a \vee (b \vee c) \equiv (a \vee b) \vee c, \; a \wedge (b \wedge c) \equiv (a \wedge b) \wedge c.$$

5. Закон дистрибутивности:

$$a \vee (b \wedge c) \equiv (a \vee b) \wedge (a \vee c)$$
$$a \wedge (b \vee c) \equiv (a \wedge b) \vee (a \wedge c).$$

6. Связь дизъюнкции и конъюнкции (закон Де Моргана):

$$\neg (a \vee b) \equiv \neg a \wedge \neg b$$
$$\neg (a \wedge b) \equiv \neg a \vee \neg b$$

7. Связь импликации с дизъюнкцией и конъюнкцией:

$$a \supset b \equiv \neg a \vee b$$
$$a \supset b \equiv \neg (a \wedge \neg b)$$

8. Закон двойного отрицания:

$$\neg \neg a \equiv a.$$

Все эти законы можно проверить с помощью таблиц истинности.

Например:

$$\neg (a \vee b) \equiv \neg a \wedge \neg b$$

a	b	$a \vee b$	$\neg (a \vee b)$	$\neg a \wedge \neg b$	$\neg a$	$\neg b$
и	и	и	(л)	(л)	л	л
и	л	и	(л)	(л)	л	и
л	и	и	(л)	(л)	и	л
л	л	л	(и)	(и)	и	и

С помощью этих законов можно осуществлять преобразования одного сложного высказывания в другое, эквивалентное первому.

Понятие о логических исчислениях

Математическая логика занимается анализом предложений, суждений и доказательств, при этом основное внимание обращается не на содержание, а на форму. Различие между формой и содержанием можно пояснить на примере двух рассуждений:

1. Москва расположена севернее Киева, Киев расположен севернее Одессы, следовательно, Москва лежит севернее Одессы.

2. Один меньше двух, два меньше трех, следовательно, один меньше трех.

Эти два рассуждения отличаются содержанием, однако имеют одну и ту же форму.

Для исследования форм рассуждений, а также зависимостей между ними вводится формализованный язык.

Опишем одно из возможных исчислений — пропозициональное (P_1).

Чтобы описать некоторый формализованный язык, нужно прежде всего задать его алфавит (перечень исходных символов) и правила образования (или распознавания) выражений этого языка.

Исходными символами исчисления P являются три несобственных символа $][\supset$, одна константа f и бесконечное число переменных $pqrsp_1q_1r_1s_1 \ldots$. Переменные и константа называются собственными символами.

Правила построения выражений исчисления P_1:

1. Стоящая отдельно исходная константа есть правильно построенная формула (*п.п.ф.*).

2. Отдельно стоящая переменная есть *п.п.ф.*

3. Если Γ и Δ суть *п.п.ф.*, то и $[\Gamma \supset \Delta]$ есть *п.п.ф.* Если формула является *п.п.ф.*, то ее единственным образом можно представить в виде $[A \supset B]$, где A есть *п.п.ф.*, она называется антецедентом, B — *п.п.ф.*, называемая консенквентом, $\supset$ — главный знак импликации. Некоторые из числа *п.п.ф.* являются аксиомами.

Правилами вывода являются следующие два:

Из $[A \supset B]$ и A следует B (правило модус поненс);

Если b — переменная, то из A следует $S_B^b A$ (правило подстановки), где $S_B^b A$ есть результат подстановки формулы B вместо всех вхождений переменной b в формулу A.

По этим правилам вывода из соответствующих *п.п.ф.* (посылок) непосредственно выводится, или непосредственно следует некоторая правильно построенная формула (заключение).

Аксиомами исчисления P_1 являются три следующие *п.п.ф.*:

$$[p \supset [q \supset p]];$$

$$[[s \supset [p \supset q]] \supset [[s \supset p] \supset [s \supset q]]];$$

$$[[[p \supset f] \supset f] \supset p].$$

Конечная последовательность, состоящая из одной или большего числа *п.п.ф.*, называется доказательством, если каждая *п.п.ф.* в последовательности либо является аксиомой, либо непосредственно выводится по одному из правил вывода из предыдущих *п.п.ф.* последовательности*. Те *п.п.ф.*, для которых существуют доказательства, называются теоремами.

С исчислением P_1 связываются правила интерпретации (семантические правила), которые задаются таблицами истинности.

---

* В частности, всякая аксиома является теоремой, доказательство которой состоит из одной единственной *п.п.ф.* — самой аксиомы.

Как и переменные в алгебре логики, переменные исчисления P_1 принимают два значения: «истина» и «ложь». Константа f — всегда одно: «ложь». Учитывая это, мы будем требовать, чтобы семантические правила, которые должны указывать интерпретацию, были бы такими, чтобы аксиомы были всегда истинными формулами.

Формула исчисления P_1 называется тавтологией, если она имеет значение «истина» для всякого набора значений ее переменных. Можно доказать, что каждая теорема исчисления P_1 является тавтологией и, наоборот, любая тавтология является теоремой исчисления P_1, т. е. выводима в исчислении P_1. Это утверждение порождает эффективную процедуру — процедуру разрешения, с помощью которой относительно любой *п.п.ф.* исчисления можно решить вопрос: является она теоремой или нет.

Эта процедура сводится к построению таблицы истинности для предъявленной *п.п.ф.* Если под главным знаком импликации* при любых распределениях истинностных значений будет стоять «истина», то предъявленная *п.п.ф.* есть тавтология, а значит, и теорема исчисления P_1. Следовательно, существует доказательство (в определенном выше смысле) этой *п.п.ф.* — теоремы в исчислении P_1.

В завершение рассмотрения исчисления P_1 приводим доказательство** *п.п.ф.* $[p \supset p]$:

1. $[s \supset [p \supset q]] \supset [[s \supset p] \supset [s \supset q]]$	аксиома 2
2. $[s \supset [r \supset q]] \supset [[s \supset r] \supset [s \supset q]]$	S_r^p [1]/
3. $[s \supset [r \supset p]] \supset [[s \supset r] \supset [s \supset p]]$	S_p^q [2]/
4. $[p \supset [r \supset p]] \supset [[p \supset r] \supset [p \supset p]]$	S_p^s [3]/
5. $[p \supset [q \supset p]] \supset [[p \supset q] \supset [p \supset p]]$	S_q^r [4]/
6. $[p \supset [q \supset p]]$	аксиома 1
7. $[p \supset q] \supset [p \supset p]$	модус поненс (5) и (6)
8. $[p \supset [q \supset p] \supset [p \supset p]]$	$S_{[q \supset p]}^q$ [7]/
9. $[p \supset p]$	модус поненс (6) и (8).

Логика предикатов

Для того чтобы иметь возможность записывать на формальном языке не только формулы, в которых переменные (которыми обозначаются элементарные высказывания) связываются с помощью логических связок, но и формулы, учитывающие структу-

---

* С помощью главного знака импликации любая *п.п.ф.* единственным образом делится на две части: левую и правую.

** Там, где это не приведет к неоднозначности, некоторые скобки будем опускать.

ру этих элементарных высказываний, вводится язык, называемый логикой предикатов. В этом языке различают предметные (индивидные) переменные и функциональные (предикатные) переменные. Предикаты могут быть одноместными, двухместными и n-местными, где n может быть 3, 4, . . ., n. Пример одноместного предиката: $P_1(x)$ — «x — простое число». Пример двухместного предиката: $P_2(x, y)$ — «x больше y». Переменные, входящие в предикат, называются предметным переменными.

Предикат всегда определен на некотором множестве предметов*, которые могут быть подставлены в предикаты вместо переменных. После такой подстановки предикат превращается в высказывание, которое может быть либо ложным, либо истинным. Поэтому предикаты можно связывать логическими связками как пропозиционные переменные.

Предикат $P_2(x, y)$ определен на множестве пар чисел натурального ряда. Для пары (5, 4) он истинен, для пары (4, 5) он ложен и т. д.

Таким образом, предикат разбивает множество, на котором он определен, на два подмножества: на одном из них он истинен, на другом — ложен. В частности, одно из этих подмножеств может быть пустым.

На языке логики предикатов можно записывать высказывания не только об отдельном объекте, но и о целом классе объектов.

В этот язык вводятся два квантора: квантор общности — $\forall (x)$ и квантор существования $\exists (x)$. Пусть $P(x)$ есть предикат. Выражение $\forall (x)P(x)$ читается так: «для всех x $P(x)$ — истина» или: «все x обладают свойством $P(x)$. Выражение $\exists (x)P(x)$ читается так: «существует такой x, для которого $P(x)$ — истина» или «существует такой x, который обладает свойством $P(x)$».

Пусть упомянутый выше предикат $P(x)$ будет предикатом — «быть четным числом», определенным на множестве чисел натурального ряда. Тогда выражение $(x)P(x)$ — ложно, т. к. среди чисел натурального ряда есть и нечетные числа, а выражение $\exists xP(x)$ — истинно, так как оно утверждает существование хотя бы одного четного числа в натуральном ряду.

Для любого предиката $P(x)$, определенного на множестве M, выражения $\exists (x)P(x)$ и $\forall (x)P(x)$–суть высказывания, а не предикаты, т. е. они могут быть либо истинными, либо ложными. Операция, связанная с применением кванторов, называется квантификацией.

В заключение рассмотрим один поучительный пример, иллюстрирующий влияние изменения порядка применения кванторов на значение высказывания: $\exists (y)$А $(x)(x>y)$ — «существует действительное число (хотя бы одно y), меньшее любого другого действительного числа», $\forall (x)\exists$ $(y)(x>y)$ — «для каждого действительного числа существует (свое) меньшее действительное число».

---

* Точнее, на множестве имен этих предметов.

Если первое высказывание ложно, то второе — истинно. Как и в случае логики высказываний, логику предикатов можно построить как исчисление (функциональное исчисление), задав алфавит, перечень аксиом и перечень правил вывода.

Элементы структурной лингвистики

Математическая лингвистика изучает естественный язык и его свойства. Это изучение проводится на разных уровнях: фонологическом, морфологическом или синтаксическом. Одной из задач этой науки является описание того, как строятся некоторые более сложные языковые объекты из более простых: для фонологического уровня — морфемы (наименьшие осмысленные единицы языка: корни, суффиксы и т. п.), из фонем. для морфологического уровня — слова из морфем, для синтаксического — предложения из слов. В рамках математической лингвистики строятся, по крайней мере, две абстрактные модели: модель говорящего (порождающая грамматика) и модель слушающего (распознающая грамматика).

Распознающая грамматика позволяет решать, является ли предъявляемая цепочка символов синтаксически правильной или нет; в случае положительного ответа на этот вопрос грамматика позволяет построить синтаксическую структуру этой цепочки.

Порождающая грамматика должна строить из заданных символов любую синтаксически правильную цепочку (фразу) с указанием ее структуры, причем не строить ни одной неправильной.

Под грамматиками в математической лингвистике понимаются некоторые специальные системы правил, задающие множества цепочек символов. Эти цепочки могут интерпретироваться как языковые объекты разных уровней.

Между обычными и формальными грамматиками имеется существенное различие. В формальных грамматиках все утверждения формулируются исключительно в терминах небольшого числа четко определенных и весьма элементарных символов и операций. Это делает формальные грамматики простыми с точки зрения их логического строения. Однако эта же особенность приводит к тому, что формальные грамматики оказываются весьма громоздкими с точки зрения их применения. Порождающая грамматика — это система, состоящая из четырех элементов: основной, или терминальный, словарь, вспомогательный словарь, начальный символ, набор правил подстановки

$$\Gamma < V_т,\ V_{вс},\ \hat{S},\ П >.$$

1. Терминальный словарь ($V_т$) — набор исходных элементов, из которых строятся цепочки, порождаемые грамматикой.

2. Вспомогательный словарь ($V_{вс}$) — набор символов, которыми обозначаются классы исходных элементов или цепочек исходных элементов.

3. Начальный символ $(\hat{S})$ — выделенный нетерминальный символ, обозначающий совокупность (класс) всех тех языковых объектов, для описания которых предназначается данная грамматика.

4. Правила подстановки (П) — выражения вида «$X \to У$», что означает «заменить X на $У$» или «подставить $У$ вместо X», где X и $У$ — цепочки, содержащие любые терминальные или нетерминальные символы.

Если имеются две цепочки X и $У$, причем $X = Z_1AZ_2$, $У = Z_1BZ_2$ и в данной грамматике имеется правило подстановки $A \to B$, то $У$ непосредственно выводимо из X (X и $У$ — смежные цепочки).

Если имеется последовательность цепочек $X_0, X_1, \ldots, X_n$, в которой каждая следующая цепочка непосредственно выводима из предыдущей, то X_n выводима из X_0.

Сама же последовательность $X_0, X_1, X_2, X_3, \ldots, X_n$ называется выводом X_n из X_0, причем X_0 есть $\hat{S}$, X_n — цепочка в словаре $V_т$. Очевидно, что выводом в грамматике является процедура последовательного осуществления подстановок вплоть до того момента, когда в цепочке останутся только терминальные символы.

Следует подчеркнуть, что порождающая грамматика не является алгоритмом: правила подстановки — это не последовательность описаний, а совокупность разрешений*. Это означает, что, во-первых, правило вида $A \to B$ понимается в грамматике как «A можно заменить на B» (а можно и не заменять), тогда как в алгоритме $A \to B$ означало бы «A следует заменить на B» (нельзя не заменить); во-вторых, порядок применения правил в грамматике произволен (неоднозначен).

Совокупность всех терминальных цепочек, выводимых из начального символа в грамматике Г, называется языком, порождаемым грамматикой, и обозначается $L(Г)$.

Рассмотрим один класс порождающих грамматик: грамматику непосредственно-составляющих (*НС*).

В грамматиках *НС* требуется, чтобы в каждом правиле вида $A \to B$ левая часть (A) имела вид Z_1cZ_2, где c — в точности один символ, и правая часть (B) — вид $Z_1\omega Z_2$, где ω — непустая цепочка. Будем истолковывать терминальные символы как словоформы (некоторого естественного языка), вспомогательные символы — как синтаксические категории (U — глагол, S — существительное, A — прилагательное, D — наречие, $\tilde{V}$ — группа глагола, $\tilde{S}$ — группа существительного), начальный символ $(\hat{S})$ как «предложение», а выводимые терминальные цепочки — как правильные предложения данного языка. Тогда вывод предложения интерпретируется как его синтаксическая структура, представленная в терминах *НС*.

Отметим некоторые недостатки метода *НС*, которые в равной степени относятся к другим грамматикам.

---

* Здесь видна аналогия грамматик с логическими исчислениями.

1. С помощью *НС* грамматик не удается, естественно, описать фразы, содержащие разрывные составляющие.

2. *НС* грамматика содержит только правильные формы языковых выражений. Однако владение языком обязательно предполагает умение не только построить правильную фразу, но и перейти от одной фразы к другой, например полностью синонимичной ей. Соответственно формальный аппарат, учитывающий эту особенность, должен включать в себя правила преобразования, или трансформации. Инвариантом всех трансформаций является смысл, иначе говоря, трансформации — это преобразования, сохраняющие (или почти сохраняющие) смысл. Трансформации относятся не к тому же уровню, что и *НС* грамматики: *НС* грамматики — к синтаксическому уровню, а трансформации — к семантическому.

Теперь приведем конкретный пример *НС* грамматики Г, строящей фразу «Маленький мальчик живет далеко».

$V_т$ — далеко, живет, маленький мальчик.

Начальный символ $(\hat{S})$ — Предл.

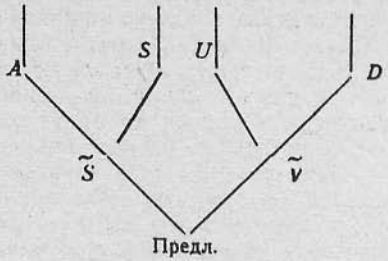
Система подстановок: Предл. $\rightarrow \tilde{S}\,\tilde{V}$, $\tilde{S} \rightarrow AS$, $\tilde{V} \rightarrow UD$,
$A \rightarrow$ маленький, $S \rightarrow$ мальчик, $D \rightarrow$ далеко, $V \rightarrow$ живет.

Приводим порождение (вывод) данной фразы (в левом столбце будет записан сам вывод, в правом — результат промежуточной подстановки).

Предл. $\rightarrow \tilde{S}\,\tilde{V}$	$\tilde{S}\,\tilde{V}$
$\tilde{S} \rightarrow AS$	$AS\,\tilde{V}$
$\tilde{V} \rightarrow UD$	$ASUD$
$A \rightarrow$ маленький	маленький SUD
$S \rightarrow$ мальчик	маленький мальчик VD
$U \rightarrow$ живет	маленький мальчик живет UD
$D \rightarrow$ далеко	маленький мальчик живет далеко

С помощью этой грамматики можно породить, в частности, и такую фразу «живет далеко маленький мальчик». Эти две фразы представляют язык $L(\Gamma)$.

Грамматики могут быть также представлены в виде скобочной структуры и в виде дерева *НС:*



ЛИТЕРАТУРА

Гладкий А. В. Формальные грамматики и языки. М., «Наука», 1973, с. 9—30.

Гладкий А. В., Мельчук И. А. Элементы математической лингвистики. М., «Наука», 1969, с. 23—49, 49—74.

Глушков В. М. Введение в кибернетику. Киев, Изд-во АН СССР, 1964, с. 87—88, 286—293.

Машины Тьюринга и рекурсивные функции. М., «Мир», 1972, с. 11—17.

Трахтенброт Б. А. Алгоритмы и машинное решение задач. § 1, 8, 9. М., «Физматгиз», 1960.

Черг А. Введение в математическую логику. М., Изд-во Иностранной л-ры, 1960, с. 66—80.

Шрейдер Ю. А. О понятии «математическая модель языка». сер. «Математика и кибернетика», № 11. М., «Знание», 1971. 64 с.

ОГЛАВЛЕНИЕ

ИНФОРМАЦИОННО-ПОИСКОВЫЕ СИСТЕМЫ
И ИНФОРМАЦИОННО-ПОИСКОВЫЕ ЯЗЫКИ

(Лекции)

Автор Б.Р. Певзнер
Ответственный за выпуск А.Н.Кусков

Изд. № 224 Подписано в печать 18.7.1977 г. Тираж 7200 экз.
Цена 6 коп. Заказ 6737

Производственно-издательский комбинат ВИНИТИ
Люберцы, Октябрьский проспект, 403