АНАЛИЗ СИМВОЛЬНЫХ ПОСЛЕДОВАТЕЛЬНОСТЕЙ
(Вычислительные системы)

1985 год Выпуск 113

УДК 621.391

РАСПОЗНАВАНИЕ ФРАЗ ДИСКРЕТНОЙ РЕЧИ С УЧЕТОМ СЕМАНТИКО-СИНТАКСИЧЕСКИХ И ПРАГМАТИЧЕСКИХ ОГРАНИЧЕНИЙ

В.Г.Лебедев, С.А.Хамидуллин

Введение

Рассматривается система, предназначенная для распознавания фраз дискретной речи.

Под фразами дискретной речи Φ_j, $j = 1 \div N$ будем понимать цепочки из отдельных высказываний S_i, $i=1 \div K$, разделенных короткими паузами. Здесь N – общее число возможных фраз при фиксированных S_i, K – объем заданного словаря.

В качестве S_i могут выступать отдельные слова, словоформы, а также короткие слитные словосочетания. Под заданным словарем будем понимать некоторое фиксированное множество $S = \{S_i\}$ (в дальнейшем для простоты будем называть элементы S_i словами). Минимальная величина пауз, разделяющих элементы S_i во фразе, должна составлять 100 мсек.

Пусть задан некоторый словарь, содержащий K слов, т.е. зафиксировано множество S, а также некоторое множество фраз $\Phi = \{\Phi_j\}$, составленных из слов заданного словаря.

Система функционирует в двух основных режимах: обучение и распознавание. В режиме обучения формируется множество эталонов $\{E_i\}$, соответствующих множеству S, $i = 1 \div K$, а в режиме распознавания каждой предъявленной для распознавания фразе F_k ставится в соответствие некоторая фраза из множества Φ.

Из всего множества фраз Φ будем рассматривать только допустимые фразы, т.е. фразы, удовлетворяющие семантико-синтаксическим и прагматическим ограничениям, принятым в системе. Синтаксические ограничения включают в число допустимых только грамматически пра-

вильные фразы. Семантические ограничения из числа синтаксически верно построенных фраз выделяют лишь фразы, имеющие смысл. Прагматические ограничения позволяют из семантически верно построенных фраз выделить лишь фразы, допустимые в конкретных ситуациях, определяемых спецификой конкретной прикладной области, в которой функционирует система распознавания фраз.

Таким образом, первой задачей, которая ставится перед системой распознавания фраз дискретной речи, является проверка предъявленной для распознавания фразы на допустимость. В случае, если фраза допустима, производится ее распознавание, результатом которого является текст распознанной фразы, если нет, система выдает сообщение о том, что фраза не понята.

Заметим, что процесс проверки контрольной фразы на допустимость совмещен в системе с процессом распознавания фразы.

Создание систем распознавания речи является актуальной задачей для многих прикладных областей, где встает необходимость речевого ввода информации в ЭВМ. Так, данная система разработана для использования на тренажере диспетчера управления воздушным движением. Здесь ставится задача автоматического понимания реплик обучающихся на тренажере курсантов с целью последующей имитации со стороны ЭВМ воздушной обстановки в районе аэропорта. Сейчас эта функция выполняется опытным пилотом-оператором.

От имеющихся ныне систем распознавания речи описываемая система отличается введением адаптивной отсечки неперспективных претендентов и более эффективной стратегией учета семантико-синтаксических ограничений по сравнению с известными [I]. Наличие семантико-синтаксических и прагматических ограничений, определяющих подмножества допустимых S_i на каждом этапе распознавания фразы F_k, выгодно отличает данную систему от систем распознавания изолированных слов. По сравнению с системами распознавания слитной речи данная система имеет ограничение, заключающееся в требовании обязательных пауз, разделяющих S_i. С другой стороны, введение этого ограничения позволяет добиться существенного сокращения трудоемкости, т.к. отпадает необходимость членения фразы на слова. Важной особенностью системы является также возможность быстрой ее модификации с помощью средств операционной системы (РАФОС или RT-II), поддерживающей функционирование программ, входящих в систему распознавания. Большинство существующих ныне систем распознавания на базе микро-ЭВМ функционируют без какой-либо операционной

системы либо созданы в виде спецпроцессоров и, следовательно, лишены этой возможности.

I. Система распознавания дискретной речи

А р х и т е к т у р а с и с т е м ы. Система реализована на микро-ЭВМ "Электроника-60" с кассетным накопителем и "электронным"*) системным диском. Она включает в себя спецпроцессор для ввода и предварительной обработки речевых сигналов и комплекс программ, написанных на языке МАКРОАССЕМБЛЕР и ФОРТРАН в операционной системе РАФОС.

Спецпроцессор позволяет выделять признаки речевого сигнала в цифровом виде (средняя интенсивность в шести полосах и число переходов речевого сигнала через нулевой уровень), усредненные за период времени T, который задается программно и обычно принимает значения, близкие к 8 мсек и 16 мсек.

Обмен данными между спецпроцессором и центральным процессором ЭВМ может осуществляться в программном режиме и в режиме прерывания программы.

Ввод речевого сигнала в спецпроцессор осуществляется непосредственно с микрофона.

В данной системе обмен осуществляется по флагу готовности.

Программное обеспечение системы включает в себя программу ввода, обучения и распознавания фраз дискретной речи и программы, служащие для предварительной подготовки словаря и формирования дерева сопрягаемости слов во фразах.

Программа ввода, обучения и распознавания осуществляет следующие функции:

- распределение оперативной памяти под эталоны E_i и контрольную реализацию, под которой понимается очередное слово из фразы F_k на этапе распознавания;
- обучение на основе данного словаря;
- распознавание фраз дискретной речи;
- запись эталонов в файл;
- замену отдельных эталонов.

---

*) "Электронный" диск - набор плат памяти, полностью эквивалентных платам оперативной памяти и имитирующих системное устройство с прямым доступом.

Весь процесс работы производится в диалоге пользователя с ЭВМ. На каждое требуемое со стороны пользователя действие программа выдает запросы-подсказки. Это дает возможность легко освоить систему широкому кругу пользователей.

Р а с п р е д е л е н и е п а м я т и. Эталоны и контрольная реализация представляются в программе в виде массивов признаков, эталонных и контрольного соответственно. Для каждого эталона предоставляется фиксированный объем памяти, который определяется на основе следующих задаваемых пользователем параметров:

- количество слов в словаре;
- желаемая длина контрольной реализации;
- максимальный размер коридора при распознавании для алгоритма динамического программирования (см. [2]);
- имя файла, содержащего дерево сопрягаемости слов во фразах на заданном наборе фраз (информация, считываемая из этого файла, занимает часть оперативной памяти).

На первом шаге программа вычисляет объем памяти для каждого эталона (оставляя желаемый размер для записи контрольной реализации) исходя из того, что исходные вектора интенсивностей, вырабатываемые спецпроцессором, не подвергаются сжатию (см.ниже). Если этот объем не устраивает пользователя, производится пересчет объема памяти при степени сжатия этих векторов интенсивностей вдвое, и т.д., пока пользователь не согласится с предлагаемым вариантом, либо степень сжатия не превысит допустимую (в системе равную пяти). Большая степень сжатия чревата существенной потерей информации об исходном сигнале. После этого можно будет повторить процесс распределения памяти с самого начала (если какой-либо промежуточный вариант кажется приемлемым) либо закончить работу.

В в о д р е ч е в о г о с и г н а л а. Через микрофон речевой сигнал поступает на спецпроцессор, который каждые 16 мсек определяет значения интенсивностей на выходах шести фильтров, перекрывающих полосу от 400 до 5000 гц. Считывание значений этих интенсивностей производится со специальных регистров при выставленном флаге готовности спецпроцессора.

По величине суммарной интенсивности, которая должна превысить заданный порог, определяется начало ввода сигнала. Окончательное решение о начале ввода принимается, если подряд несколько вводимых сегментов удовлетворяют этому условию. Решение об окончании ввода принимается, если подряд несколько вводимых сегментов

имеют суммарную интенсивность ниже пороговой. В противном случае сегменты с низкой интенсивностью соответствуют речевой паузе. Значение порога и необходимое число сегментов в первом и во втором случаях задаются пользователем.

С е г м е н т а ц и я. Параллельно с процессом ввода исходных векторов интенсивности, вырабатываемых спецпроцессором, производятся:

- замена заданного числа входных векторов интенсивностей одним усредненным;

- объединение уже усредненных близких между собой смежных векторов интенсивности в группы (сегменты) и их последующее усреднение (сегментация).

Вторичное усреднение производится для группы векторов, расстояние между первыми из которых и всеми последующими меньше так называемого порога сегментации, тоже задаваемого пользователем.

П о с т р о е н и е п р и з н а к о в. После проведения сегментации строятся вторичные признаки $P_l = 256(R_l+G_l)/(Q+D)$, $l = 1 \div 6$, где R_l - значение усредненной интенсивности в l-й полосе после сегментации, $Q=\sum_l R_l$, G_l и D - регуляризирующие добавки, тоже задаваемые программно. Они вводятся, чтобы несколько нивелировать случайные шумовые добавки к речевому сигналу в паузах.

Таким образом, по окончании программы ввода имеется массив признаков речевого сигнала, каждый из шестиэлементных сегментов которого соответствует набору значений с выхода шести фильтров.

Р е ж и м о б у ч е н и я состоит в том, что диктор произносит в микрофон последовательно каждое слово S_i из заданного словаря. На каждое произнесенное слово вырабатывается массив признаков, называемый эталоном данного слова (E_i).

В этом режиме на экране терминала высвечиваются одно за другим слова из заранее подготовленного словаря. После произнесения каждого слова следует запрос программы на повторение ввода этого же слова, либо продолжение обучения. В зависимости от ответа пользователя можно выбрать один из вариантов или же приостановить процесс обучения (если не нажимать ни одну из клавиш).

После окончания обучения имеется полный массив эталонов и массив их длин. Программа готова к режиму распознавания.

Запись эталонов в файл служит для запоминания эталонных реализаций с тем, чтобы при последующих сеансах пользователь не проводил режим обучения заново, а лишь прочитал файл с заранее построенными эталонами.

При записи требуется задать имя файла, куда будут записаны массив эталонов и массив их длин.

Следует заметить, что при последующем считывании эталонов требуется задать в точности то же распределение памяти под эталоны и, желательно, те же параметры, определяющие сегментацию и пороги для ввода речевого сигнала.

Замена эталонов. Режим замены эталонов позволяет обновить отдельные неудачно введенные на предыдущем этапе обучения (или замены) эталоны. Для их обновления указывается номер заменяемого слова, все остальные действия аналогичны режиму обучения.

Распознавание слов во фразах производится с помощью метода динамического программирования с адаптивным коридором [2]. Рассмотрим процесс распознавания некоторого слова S_m (контрольной реализации), входящего в состав контрольной фразы F_k.

Решающее правило имеет вид: $n = \arg\min_i R_i$, $i=1 \div K$, где R_i – расстояния S_m до эталонов E_i, вычисленные с помощью алгоритма динамического программирования; n – номер эталона, принимаемого в качестве решения о слове S_m.

Пусть S_m состоит из l сегментов. Эти сегменты последовательно поступают на вход алгоритма распознавания слов. При поступлении очередного сегмента с номером j $(j=1,2,\ldots,l)$ производится переопределение всех R_i $(i=1,2,\ldots,K)$. Такая организация вычисления R_i позволяет в процессе распознавания S_m проводить отсечку малоперспективных эталонов.

Условие отсечки имеет вид: если $R_i/R_{min} > B$ (где $R_{min} = \min_i R_i$), то эталон E_i отсекается как малоперспективный в смысле его вероятности быть принятым в качестве решения относительно слова S_m (здесь B – порог отсечки).

Отсеченные эталоны не участвуют в последующих вычислениях R_i до окончания распознавания S_m.

Порог отсечки B монотонно уменьшается с ростом j – номера сегмента контрольной реализации. При $j=1$ порог отсечки $B = B_0$, при $j=h$ – $B = 2$, при $j = 3h-1$ – $B = (8h+1)/(7h-1)$. Далее по-

рог отсечки остается постоянным. Значения параметров B_0 и h задаются пользователем.

Такое поведение порога отсечки позволяет на первых шагах исключить из рассмотрения эталоны, наиболее отличающиеся от контрольной реализации своими начальными сегментами. По мере роста номеров контрольной реализации учитываются все более тонкие различия. Распознавание заканчивается, если на каком-то шаге остается лишь один претендент. В противном случае по окончании вычисления расстояний выбирается эталон, имеющий минимальное расстояние до контрольной реализации.

В систему введено также у с л о в и е о т к а з а от распознавания, которое формулируется следующим образом: если $R_{min} > R^*$ (где R^* - значение порога отказа, задаваемое пользователем), то система отказывается от распознавания слова S_m.

В целом процесс распознавания фразы F_k выглядит следующим образом.

После задания всех порогов контрольная фраза пословно произносится в микрофон. По окончании распознавания каждого произнесенного слова на экран выводится распознанное слово, его номер в словаре, расстояние до контрольной реализации, номер второго претендента, его расстояние до контрольной реализации и время распознавания.

Если расстояние от первого претендента до контрольной реализации превысит порог отказа, то программа вместо текста распознаваемого слова напечатает сообщение "к сожалению, Вас не понял" (а также номера претендентов, расстояния и время распознавания), после чего распознавание текущей фразы прекращается. Система перейдет, в зависимости от ответа пользователя, к распознаванию новой фразы, либо изменит режим работы.

У ч е т с е м а н т и к о - с и н т а к с и ч е с к и х и п р а г м а т и ч е с к и х о г р а н и ч е н и й. Процессу распознавания предшествует формирование допустимых фраз, т.е. последовательностей слов, удовлетворяющих семантико-синтаксическим и прагматическим ограничениям. После этого в процессе распознавания фразы на очередном шаге будут рассматриваться только эталоны допустимых на данном шаге слов.

Удобно такие допустимые последовательности слов во фразах представить в виде дерева, каждая ветвь которого отображает продолжение допустимой фразы.

По сравнению с известными методами [I], где ограничения на слова во фразе строятся путем задания наборов слов, допустимых при распознавании очередного элемента фразы, ограничения в виде дерева позволяют сократить перебор, так как на очередном шаге распознавание слова производится на подсловаре, допустимом только для данного элемента данной фразы, а не всех фраз, как это имеет место в предыдущем случае.

Задание ограничений в виде матрицы сочетаемости слов требует по сравнению с предлагаемым методом гораздо большего объема оперативной памяти. Так, для словаря из 100 слов требуется хранить матрицу сочетаемости слов, содержащую 10000 элементов.

Формирование дерева сопрягаемости слов во фразах. Таким образом, рассматривается последовательность допустимых фраз. Из вершины первого уровня (начала распознавания) мы можем попасть в любую вершину, соответствующую одному из слов, стоящих на первом месте во фразах, от них ветви ведут к допустимым в продолжениях каждой из фраз словам и т.д.

В программе такое дерево строится в виде двумерного массива, объединяющего последовательность блоков, каждый из которых является набором допустимых узлов (номеров слов), имеющих связь с узлом предыдущего уровня (первая строка блока) и ссылочных адресов на блоки последующего уровня (вторая строка). Ссылочные адреса, равные нулю, определяют конец фразы. Начинается каждый блок идентификатором блока (-I или -2 в первой строке) и числом элементов блока (во второй строке). Программа формирования дерева сопрягаемости слов во фразах работает в диалоговом режиме и позволяет вводить исходные данные, определяющие последовательность слов во фразе, с клавиатуры терминала либо из ранее подготовленного внешнего файла. Результатом деятельности программы является выходной файл, содержащий требуемый массив сопрягаемости и готовый к использованию в программе распознавания.

Входные данные, описывающие фразы, представляются в виде последовательности строк, каждая из которых описывает конкретную фразу (группу фраз) или ее часть и имеет вид:

$$[j] \; A_1, A_2, \ldots, A_k \; [\cdot],$$

где j – номер текущей фразы, необязательный элемент строки, указывающий нумерацию фразы, может использоваться для большей удобо-

читаемости текста файла; A_i - номер или набор номеров слов, могущих стоять во фразе на i-м месте (во втором случае номера должны быть обрамлены круглыми скобками); "*" - символ продолжения фразы в следующей строке, если ее описание не помещается в одну строку; "," - разделитель между соседними элементами фразы, может опускаться, если стоит рядом со скобкой.

Формат ввода данных достаточно свободный, лишние пробелы внутри скобок игнорируются при обработке строк и служат для большей удобочитаемости.

ПРИМЕР:

```
I.  (I0,I2), II, (I,2,3,4,5,6,7,8,*
      9), (I3,I4)
2.  (25)24(23,2I)(I3,I4)
3.  (I5)(29)
4.  (I5)(92) .
```

Первая и вторая строки приведенного файла описывают 36 фраз, в которых на первом месте могут быть I0-е или I2-е слова, на втором - II-е, на третьем - I-е, 2-е,..., или 9-е, на четвертом - I3-е или I4-е слова из некоторого фиксированного словаря, символ "*" указывает на продолжение фразы во второй строке.

Третья строка описывает 4 фразы, в которых на первом месте может быть 25-е слово, на втором - 24-е, на третьем - 23-е или 2I-е, на четвертом - I3-е или I4-е.

Четвертая строка описывает фразу с I5-м словом на первом месте и 29-м на втором.

Пятая строка описывает фразу с I5 словом на первом месте и 92-м на втором.

Начальные фрагменты строк: "I.", "2.", "3." и "4" служат для лучшего восприятия текста файла (используются для нумерации фраз или, точнее сказать, групп фраз). Следует отметить, что третью и четвертую фразы можно представить в виде (I5)(29,92), что более компактно и более наглядно.

О п т и м и з а ц и я. Учитывая специфику области - большое количество идентичных ветвей - программно производится удаление повторяющихся ветвей с заменой ссылок на удаленные узлы ссылками на оставшиеся аналогичные.

Массив фраз, построенный по приведенному выше примеру, имеет вид:

I	2	3	4	5	6	7	8	9	I0	II	I2	I3	I4
28	I0	I2	25	I5	-2	II	-2	I	2	3	4	5	6
4	6	6	2I	26	I	8	99	I8	I8	I8	I8	I8	I8
I5	I6	I7	I8	I9	20	2I	22	23	24	25	26	27	28
7	8	9	-2	I3	I4	-I	22	-I	23	2I	-I	29	92
I8	I8	I8	2	0	0	I	23	2	I8	I8	2	0	0

Первоначально построенный массив, описывающий дерево сопряжаемости слов во фразах, занимал бы 306 слов вместо 56, как это имеет место после проведения оптимизации.

Первая строка служит для относительной нумерации элементов массива. Вместо признака первого блока (-I в первом элементе первой строки) записывается полная длина массива.

3. Заключение

Испытания системы проводились на фразах проблемно-ориентированного словаря, представляющих собой фразы языка диспетчера управления воздушным движением. Объем словаря - I20 слов. На материале из I40 фраз при подстройке под диктора получена надежность распознавания слов 98%. Следует отметить сложность словаря, в состав которого входят очень близкие слова или словосочетания: аб - ав - аг - ад, бб - бв - бг - бд, место - метео, два - две - надцать- две - двадцать и т.д. Коэффициент ветвления изменяется в пределах от I до 48 и в среднем равен I3. Система работает в реальном масштабе времени. В настоящее время система находится в опытной эксплуатации.

Литература

I. Опыт речевого управления вычислительной машиной /Г.Я.Высоцкий, Б.Н.Рудный, В.Н.Трунин-Донской, Г.И.Цемель - Изв. АН СССР, Техническая кибернетика, I970, №2, с.I34-I43.

2. SAKOE H., CHIBA S. Dynamic Programming Algorithm Optimization for Spoken Word Recognition.- IEEE Trans.Acous.Speech and Signal Processing,1978,v.ASSP-26,N 1,p.43-49.

Поступила в ред.-изд.отд.
29 ноября I985 года