АНАЛИЗ СИМВОЛЬНЫХ ПОСЛЕДОВАТЕЛЬНОСТЕЙ
(Вычислительные системы)
1985 год Выпуск 113

УДК 621.391:534.4:691.3.06:51

МИНИМИЗАЦИЯ ВЫЧИСЛЕНИЙ В РАСПОЗНАВАНИИ РЕЧИ

В.М. Величко

При построении систем распознавания и понимания слитной речи на базе микрокомпьютеров возникают проблемы сокращения объема вычислений, необходимых для принятия решения о произнесенной фразе. Проблемы вызваны как большим перебором вариантов при многообразии допустимых сочетаний слов во фразах по сравнению с распознаванием изолированных команд, так и меньшим быстродействием универсальных микрокомпьютеров по сравнению с большими ЭВМ и спецпроцессорными устройствами. В любом случае такие проблемы возникнут при увеличении словаря и ослаблении языковых ограничений.

В статье рассматриваются некоторые конкретные способы уменьшения числа слов-претендентов на разных этапах работы системы распознавания и приводятся результаты экспериментальной проверки этих способов.

При выборе способов экономии вычислений использовались следующие возможности:

а) экономия за счет предварительного отбора слов по длительности;

б) экономия за счет различия интегральных характеристик слов;

в) экономия за счет рациональной схемы вычислений-применения локальных методов оптимизации при оценке расстояния между реализациями слова;

г) экономия за счет отсечки бесперспективных продолжений фразы по глобальным и локальным критериям оценки расстояния между реализациями слов.

Предполагается использование этих способов в системе, построенной на алгоритме распознавания слитной речи [1], но они могут успешно применяться и при распознавании изолированных команд.

I. Краткая характеристика системы

Экспериментальная проверка проводилась на проблемно-ориентированном словаре диспетчера цифрового диспетчерского тренажера в режиме распознавания изолированных команд с подстройкой под диктора. Характеристики словаря существенны при использовании пунктов "а" и "б", менее существенны два пункта "г" и безразличны для пункта "в". Словарь содержит 129 слов, в том числе 62 служебных слова и 67 числительных от 0 до 1000, из них для 1-9, кроме количественных числительных, включены еще и порядковые. Самое короткое слово "до", самое длинное словосочетание "курс 275". Встречаются семейства близких по звучанию слов - "глиссада-глиссаду-глиссады-к глиссаде-на глиссаде", "курс-курса-на курс - на курсе", "метра-метров", "градуса-градусов", "тысяча-тысячи - тысячу-тысяч" и некоторые другие.

Обучающая выборка состоит из набора однократно произнесенных команд-эталонов словаря.

Обучающая выборка и контрольные реализации вводятся в микро-ЭВМ "Электроника-60" через микрофон и гребенку из 6 аналого-цифровых фильтров в диапазоне от 400 до 5000 гц, интенсивности (суммы модулей отсчетов) речевого сигнала в полосах замеряются каждые 16 мсек. Числовые значения признаков (в диапазоне от 0 до 127) нормируются на корень из суммы компонент 6-мерного вектора признаков с ограничениями от 0 до 255. Нормировка на корень из суммы вместо традиционной нормировки на сумму позволяет сгладить влияние вариаций уровня громкости, но тем не менее учитывает разницу в уровнях. Метрика из соображений экономии вычислений выбрана в виде суммы модулей разностей между компонентами 6-мерных векторов, характеризующих 16-миллисекундные сегменты команд обучающей выборки и контрольной реализации. Кроме того, для устранения необходимости нормировки по длине слова (и, следовательно, операции деления) вводится отрицательная постоянная добавка p к принятой мере расстояния [2]. В результате расстояния g_{is} между i-м сегментом контрольной реализации $\vec{x}^{(i)}$ и s-м сегментом эталона $\vec{E}^{(s)}$ определяются по формуле:

$$g_{is} = \sum_{k=1}^{6} |x_k^{(i)} - i_k^{(s)}| - p \,. \qquad (1)$$

Рассмотрим по порядку реализацию и эффективность каждой из вышеперечисленных возможностей сокращения объема вычислений.

2. Использование длительности команд

Вариация длин отдельных команд от реализации к реализации, даже с учетом ошибок в определении границ речевого сигнала, не бывает слишком большой. Поэтому были приняты следующие ограничения. Из числа эталонов-претендентов сразу исключаются эталоны, отличающиеся по длительности больше чем в VMIN раз в меньшую и больше чем в VMAX раз в большую сторону от контрольной реализации команды длиной L_c сегментов, т.е. остаются претенденты с длительностями $L_{эт}$: $L_c \times VMIN \leq L_{эт} \leq L_c \times VMAX$. Технически это осуществляется следующим образом. В процессе подготовки системы к работе при загрузке обучающей выборки производится сортировка эталонов по длительности с оглавлением для каждой группы одинаковых длительностей. Коэффициенты $L_{min} = L_c \times VMIN$ и $L_{max} = L_c \times VMAX$ однозначно определяют исходный список эталонов-претендентов, не требующий дальнейшей проверки на длительность на следующих этапах распознавания. Эффективность этого приема следующая. При многократных распознаваниях от микрофона подобраны пороги VMIN = 0,6 и VMAX = 1,7, которые гарантируют (для данного диктора) включение "своего" эталона в число претендентов. При этом среднее число претендентов составляет 79-82 (62%), а максимальное-104 (81%) из 129. Таким образом, несложный и фактически не требующий дополнительных затрат прием позволяет в среднем сократить число претендентов (а, следовательно, и операций) примерно на 40%. Для более однородных по длительности словарей экономия будет, естественно, меньше. Например, в экспериментах [3] 1967-1968 гг. со словарями 150-200 слов средняя экономия была около 15%.

3. Использование интегральных характеристик

Для отсечки бесперспективных эталонов по интегральным характеристикам слов проверялись следующие параметры: а) средние значения признаков на длине всего слова или его частей; б) среднеквадратичные отклонения признаков на длине всего слова; в) среднние абсолютные отклонения на длине всего слова; г) сумма взвешенных параметров пп. "а" и "в" на длине слова. Охарактеризуем их эффективности.

Средние значения признаков X_i (i = 1,6) на участке слова от сегмента $l_н$ до $l_к$ (на всем слове $l_н = 1, l_к = L_c$) определяются как

$$x_i = \frac{1}{l_к - l_н + 1} \sum_{j=l_н}^{l_к} x_{ij}, \qquad (3)$$

где x_{ij} – значение i-го признака для j-го сегмента.

Сформированный таким способом вектор $\vec{X}$ отражает обобщенные характеристики слова и может служить для предварительного отбора эталонов-претендентов. Для каждого эталонного слова k формируются усредненные значения $\vec{E}^{(k)}$, запоминаются вместе с обучающей выборкой и сравниваются в процессе распознавания с соответствующим средним вектором на участке контрольного слова $\vec{X}$ в метрике (I) при p = 0.

Типичные результаты выглядят следующим образом. "Свой" эталон стоит на первом месте в 80 случаях из 129 (62%) для словаря-современника, т.е. записанного с небольшим временным интервалом от процесса распознавания, и в 70 (54%) для ранее записанного словаря – с временным интервалом более полугода. При упорядочении расстояний от контрольного слова до всех эталонов "свой" эталон занимал места с I по 26, причем места с I по 9 занимали 126 слов, с I по 12 – 127, с I по 13 – 128 для словаря-современника. Для ранее записанного словаря соответствующие показатели такие: с I по II – 124 слова, с I по 15 – 127, с I по 16 – 128, с I по 30 – 129. Максимальное отношение расстояния от "своего" эталона $R(\vec{E}^{(k)}, \vec{X})$ к расстоянию до эталона, стоящего на первом месте $R(\vec{E}_1, \vec{X})$, 2,26 для словаря-современника и 2,8 для ранее записанного словаря. Максимальная разность между расстоянием до "своего" эталона и до эталона на первом месте $R(\vec{E}^{(k)}, \vec{X}) - R(\vec{E}_1, \vec{X})$ соответственно 87 и 49. Максимальная абсолютная величина расстояния от "своего" эталона $R(\vec{E}^{(k)}, \vec{X})$ соответственно 222 и 101. Приведенные числовые результаты конкретных экспериментов показывают возможности отсечки эталонов-претендентов по различным порогам – по общему числу M эталонов-претендентов, упорядоченных по расстоянию от вектора контрольной реализации; по отношению расстояния от эталона до контрольной реализации к расстоянию от эталона, ближайшего к контрольной реализации КО; по разности расстояний от контрольной реализации до эталона-претендента и до эталона, ближайшего к контрольной реализации КР; по абсолютному расстоянию от эталона до контрольной реализации КТ. В результате был сформулирован комплексный критерий отсечки следующего вида. Оставляется не более M первых эталонов-претендентов (упорядоченных по увеличению расстояния до контрольной реализации), удовлетворяющих соотношению $R(\vec{E}^{(k)}, \vec{X}) \leq$

$\leq \min(R(\vec{E}_1, \vec{X})$ *КО+КР,КТ*). Комбинация М, КО, КР и КТ подбирается экспериментально. В частности, хорошие результаты по минимизации количества эталонов-претендентов без потери надежности дают следующие комбинации: М = 29, КО = I, КР = I00, КТ = 200 или КО = = 3, КР = 30 при тех же М, КТ. В среднем после отсечки по КО, КР, КТ (без М) остается 24–30 эталонов-претендентов при максимуме до 66. С учетом М среднее равно 20 при максимуме М = 29.

Проверялись также возможности отсечки при делении слова на два и три равных интервала с двумя и тремя векторами интегральных характеристик соответственно. Оказалось, что "хорошие" эталоны в среднем становятся при этом ближе к первому месту, а "плохие" – дальше, что увеличивает вероятность отсечки "своего" эталона и показывает нецелесообразность дробления слова. Аналогичные качественные результаты получились при усреднении первых l сегментов ($l_н$ = I, $l_к$ = l) при l = 3,5,7,I0,I3,I5.

Вектор, характеризующий среднеквадратичные отклонения признаков на длине всего слова, определяется как

$$\sigma_i = \sqrt{\frac{\sum_{j=1}^{L_c} (x_{ij} - x_i)^2}{L_c - 1}},$$

где обозначения прежние.

Вектор средних абсолютных отклонений на длине слова определяется как

$$a_i = \frac{1}{L_c} \sum_{j=1}^{L_c} |x_{ij} - x_i|$$

и может служить оценкой среднеквадратичного отклонения с соответствующим весовым коэффициентом. Проверялась возможность отсечки эталонов-претендентов по тем же критериям, что и для средних значений, как по вектору $\vec{\sigma}$, так и по вектору $\vec{a}$. Результаты получились приблизительно одинаковыми, но вычисление $\vec{a}$ требует значительно меньших вычислительных затрат. В общем случае лучшую оценку для $\vec{\sigma}$ дает вычисление $\vec{a}$ с отклонением не от среднего, а от медианы. Но проверка на речевом материале показала меньшую стабильность этой оценки, что, по-видимому, связано с наличием квазистационарных участков в речевом сигнале и, следовательно, резким нарушением требования нормальности распределения величин признаков для получения оценки $\vec{\sigma}$ по отклонению от медианы.

Для удобства дальнейшего применения компоненты вектора средних абсолютных отклонений брались с коэффициентом 2. Применялась та же методика отсечки, что и по средним значениям на длине слова. Типичные результаты следующие. "Свой" эталон стоит на первом месте в 84 случаях из 129 (65%) для словаря-современника и в 54 случаях (47%) для ранее записанного словаря. При упорядочении расстояний от контрольного слова до всех эталонов "свой" эталон для словаря-современника занимает места с I по I7, причем места с I по I0 - I27 слов, с I по I6 - I28 слов. Для ранее записанного словаря эти показатели такие: с I по I0 - II6 слов, с I по I4 - I22 слова, с I по 33 - I28 слов, с I по 43 - I29 слов. Максимальное отношение расстояния от "своего" эталона к расстоянию от эталона на первом месте I,88 для словаря-современника и 2,93 для ранее записанного словаря. Максимальная разность между расстоянием до "своего" эталона и до эталона на первом месте соответственно 46 и 59. Максимальная абсолютная величина расстояния от "своего" эталона соответственно 8I и II4. Действуя по аналогии с отсечкой по средним, получаем при тех же значениях М, КО, КР и КТ в среднем 30-35 эталонов-претендентов при максимуме до 80 без учета М. С учетом М среднее равно I7 при максимуме М = 29. Интересно отметить, что пересечение эталонов-претендентов, оставшихся после отсечки по среднему и по отклонению, составляет I5-20 при максимуме 50, т.е. отсечка по средним и по отклонениям отбирает разные эталоны-претенденты. Это привело к привлечению смешанного критерия - сумме расстояний между средними $\vec{X}$ и удвоенными отклонениями $2x\vec{\sigma}$ (отсюда появился коэффициент 2 при вычислении вектора средних абсолютных отклонений). Смешанный критерий (физический смысл которого неясен) характеризуется следующими числами по аналогии со средними и отклонениями. "Свой" эталон на первом месте в 96 (74%) и 88 (68%) случаях для словаря-современника и ранее записанного словаря соответственно. Распределение мест соответственно с I по 8 - I28, с I по I5 - I29 и с I по 5 - I20, с I по 8 - I26, с I по I5 - I29. Отношение расстояний соответственно 2,08 и I,9I. Разность расстояний до "своего" и первого эталонов соответственно II7 и 69. Максимальная величина расстояний до "своего" эталона 287 и I74. Приемлемые сочетания пороговых величин для смешанного критерия следующие: М = 29, КО = I, КР = I50, КТ = 300 или КО = 2, КР = = 30. При этом среднее число остающихся эталонов-претендентов II-25 при максимуме 50 без учета М и II при максимуме М = 29. Применяя последовательно все три критерия отсечки, получаем сокраще-

ние числа эталонов-претендентов в среднем до 6-II при максимуме 43 без учета М и 8 при максимуме М = 29. Применение трехступенчатого критерия отсечки эталонов-претендентов даже увеличивает надежность распознавания, отсекая похожие в акустическом отношении слова. Например, не встречаются ошибки типа "два-ноль", "тысяч-тысяча, тысячи, тысячу", "ноль-ноля",

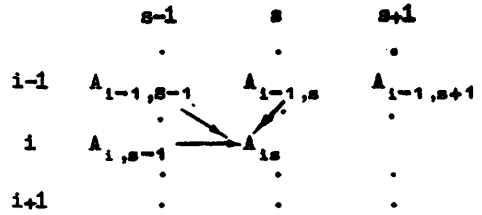
4. Схема локального динамического программирования с адаптивным коридором

Рассмотрим возможность сокращения вычислений за счет более рациональной схемы динамического программирования, а именно, применения локальной оптимизации при оценке расстояния между конт - рольной и эталонной реализациями слов.

За основу взята симметричная схема динамического программи - рования, аналогичная [3], с функционалом, минимизирующим расстояние и определяемым рекуррентной формулой

$$A_{is} = \min(A_{i-1,s}, A_{i,s-1}, A_{i-1,s-1}) + \varepsilon_{is} ,$$

где A_{is} - значение функционала для i-го сегмента контрольного слова и s-го сегмента эталонного слова, ε_{is} - расстояние из (I) (см. рисунок).



Существующие методы локальной минимизации наиболее эффективны при использовании фиксированного коридора небольшой ширины вдоль диагонали матрицы расстояний (ε_{is}) (например, [4]) или метода градиентного спуска (например, [5]). Фиксированный коридор неудобен при неизвестных границах слова в слитной фразе, а градиентный спуск может необратимо увести от оптимального пути при не очень больших локальных отличиях между эталоном и контрольной реализацией. В настоящей работе предлагается модификация динамического программирования с адаптивным коридором (ширины 3), в значительной степени ослабляющая отмеченные недостатки. Путь мини - мальной длины ищется в окрестностях точки (i,s) , доставляющей ми-

нимум функционала по сравнению с соседней точкой, лежащей на пути с тем же числом шагов от начала (в данном случае с точкой (i-1, s+1)). Требование сравнения на путях с равным числом шагов весьма существенно, так как локальные экстремумы могут быть смещены просто за счет добавления лишнего шага пути и, следовательно, лишнего расстояния ε_{is}.

Предлагаемая схема динамического программирования выглядит следующим образом. Сравниваются значения функционалов A_{is} и $A_{i-1,\,s+1}$ (число шагов для них от любой точки $(i_{нач},1)$ до точек (i,s) и $(i-1,s+1)$ одинаково). При $A_{is} \le A_{i-1,s+1}$ локально оптимальный путь ищется в точках $A_{i+1,s-1}$, $A_{i+1,s}$, $A_{i+1,s+1}$, т.е. в коридоре шириной 3 вокруг точки s для следующего значения i. В противном случае - в точках $A_{i-2,s+2}$, $A_{i-1,s+2}$, $A_{i,s+2}$ (опять в коридоре шириной 3, но вокруг точки i-1 для следующего значения s). Помимо очевидной экономии в вычислениях ($\sim 3 \times (L_C + L_{ЭТ})$ вместо $L_C \times L_{ЭТ}$ при полном динамическом программировании) описанная схема динамического программирования требует значительно меньше памяти по сравнению с полным динамическим программированием. При переходе на i+1 строку требуется помнить 4 значения функционала: $A_{i,s-1}$, $A_{i,s}$, $A_{i,s+1}$, $A_{i-1,s+1}$ для каждого эталона-претендента, дающие возможность продолжить поиск оптимального пути как по горизонтали матрицы расстояний (увеличивая s), так и по вертикали (увеличивая i). Для полного же динамического программирования необходимо помнить $L_{ЭТ}$ значений функционала.

Для сравнения был реализован метод градиентного спуска в модификации, допускающей переход из точки (i,s) в точку

$$\mathrm{argmin}(g(i,s+1),\ g(i+1,s),\ g(i+1,s+1)),$$

где $g(i,s) \equiv \varepsilon_{is}$, argmin - значение аргумента, доставляющего минимум заданному выражению.

Предложенный метод локального динамического программирования по трудоемкости в ~ 2 раза превышает указанную модификацию, но его надежность для ранее записанного словаря существенно превышает надежность модификации метода градиентного спуска (для словаря-современника разница невелика, хотя динамическое программирование и в этом случае надежнее. Просмотр многочисленных сравнений на дисплее вручную не обнаружил случаев отклонения схемы локального динамического программирования от глобально-оптимального пути (ясно, что достаточную статистику набрать таким способом невозможно, а автоматическое сравнение с полным динамическим программированием не проводилось).

5. Критерии отсечки по длине оптимального пути

Возможность сокращения вычислений за счет глобальных критериев Δ расстояния между реализациями слов рассмотрен в [2]. Там предлагается на каждой строке i матрицы функционалов A_{is} определять минимальное значение A_{opt} функционала и исключать как бесперспективные эталоны-претенденты, на которых оптимальное значение функционала A'_{opt} более чем на Δ превышает A_{opt} на всей строке. К этому критерию целесообразно добавить локальный критерий ε, по которому исключаются эталоны-претенденты, имеющие большое ($>\varepsilon$) приращение функционала A_{is} на небольшом числе шагов (в реализованном алгоритме - на 2 шагах) оптимального пути. Сочетание отсечек по Δ и ε позволяет достаточно быстро отсечь значительное число эталонов-претендентов, оставшихся после отсечек по темпу, средним и отклонениям. Как правило, алгоритм динамического программирования не доходит до конца при распознавании изолированных команд, потому что после отсечек по всем перечисленным критериям остается единственный эталон-претендент, объявляемый результатом распознавания.

6. Результаты и выводы

Типичные примеры распознавания словаря в 129 слов тестовой системой для ранее записанного словаря выглядят следующим образом. Для порогов отсечки $\Delta = \min(500+100xi, 2000)$, где i - номер сегмента контрольной реализации, ε = 600, КО = 1,1,1 соответственно для отсечки по средним, отклонениям и их сумме, КР = 100, 100, 150, КТ = 220,220,350, М = 29, допущена одна ошибка при отсечке своего эталона одновременно по Δ и ε и 4 ошибки при сравнении оптимального значения "своего" и "чужого" функционала. Надежность 96%, среднее время распознавания 1,98 сек/слово. При значениях порогов отсечки $\Delta = \min(300+100xi, 1500)$, ε = 500, КО = 1,1,1, КР = 100,100,150, КТ = 200,200,300, М = 29 допущена одна ошибка при отсечке по темпу (не опознано начало слова "сто"), 3 ошибки при отсечке "своего" эталона по Δ и 2 ошибки при сравнении функционалов. Надежность 95%, среднее время распознавания 1,55 сек/слово. Программа распознавания реализована на языке ФОРТРАН в РАФОСе (для ориентировки - усреднение параметров слова длиной 35 сегментов занимает 100 мсек), время счета включает проверку причин всех отсечек и набор статистики ошибок для однократного произнесения словаря.

Полученные результаты свидетельствуют об эффективности предложенных способов экономии вычислений с сохранением достаточно высокой для выбранного словаря надежности.

Автор выражает благодарность коллегам из речевой группы НГУ и ИМ СО АН СССР под руководством Н.Г.Загоруйко за аппаратурную и программную поддержку системы ввода и первичной обработки информации, постоянное обсуждение методов и результатов работы, а также членам речевого коллектива КБ завода "Россия" под руководством А.Н.Петрова за полезные дискуссии и обсуждения.

Литература

1. ВЕЛИЧКО В.М. Алгоритм распознавания слитной речи с использованием семантико-синтаксических ограничений.-В кн.: Автоматическое распознавание слуховых образов (АРСО-12). Тез. докл. и сообщений 12-го Всесоюз.семинара, Киев, 1982, с.342-345.

2. ВЕЛИЧКО В.М. Алгоритмы распознавания дискретной и слитной речи.-В кн.: Автоматическое распознавание слуховых образов (АРСО-13). Тез. докл. и сообщений 13-й Всесоюз. школы-семинара, ч. 2, Новосибирск, 1984, с.118-119.

3. ВЕЛИЧКО В.М., ЗАГОРУЙКО Н.Г. Автоматическое распознавание ограниченного набора устных команд.-В кн.: Вычислительные системы. Вып. 36. Новосибирск, 1969, с. 101-110.

4. ВЕЛИЧКО В.М., ЗАГОРУЙКО Н.Г. Распознавание 200 устных команд. -В кн.: Труды Акустического института. М., 1970, вып. XII.

5. ТУРКИН В.Н. Распознавание речевых образов с использованием метода градиентного спуска. -В кн.: Автоматическое распознавание слуховых образов (АРСО-13). Тез. докл. и сообщений 13-й Всесоюз. школы-семинара, ч.2, Новосибирск, 1984, с.120-121.

Поступила в ред.-изд.отд.
6 декабря 1985 года