МЕТОДЫ АНАЛИЗА ДАННЫХ
(Вычислительные системы)
1985 год
Выпуск III

УДК 621.391:518.5

ПОСТРОЕНИЕ АЛГОРИТМОВ РАСПОЗНАВАНИЯ ДИСКРЕТНОЙ И СЛИТНОЙ РЕЧИ НА ОСНОВЕ МЕТОДА ГРАДИЕНТНОГО СПУСКА

В.Н. Туркин

В настоящее время практически все методы распознавания речевых образов основаны на вычислении определенных каким-либо способом оценок мер сходства образов. Предполагается, что одним из главных факторов изменчивости речевых образов является изменение темпа речи, поэтому при сравнении необходимо осуществлять нелинейную нормализацию образов по длительности. Наиболее распространенным путем решения задачи вычисления меры сходства образов с учетом нормализации по длительности является сведение ее к задаче нахождения экстремума аддитивного функционала, которая успешно решается методом динамического программирования. В то же время известно, что метод динамического программирования требует очень больших вычислительных затрат. В настоящей работе рассмотрен иной подход, основанный на вычислении мер сходства образов методом градиентного спуска, требующий меньше вычислительных затрат. На основе предложенных алгоритмов разработана система распознавания дискретной речи [I].

I. Постановка задачи распознавания дискретной речи

Рассмотрим сначала постановку задачи распознавания речевых образов, описываемых непрерывными функциями времени. Пусть $\mathfrak{E} = \{E^1(\tau_1), E^2(\tau_2), \ldots, E^p(\tau_p), \ldots, E^P(\tau_p)\}$ – множество эталонов, где $E^p(\tau_p)$ – непрерывные вектор-функции признаков речевого сигнала в пространстве R^N, $E^0(\tau_0)$ – образ, подлежащий распознаванию.

Определим в фазовом пространстве $T_p: \{0 \le \tau_p \le T_p'; 0 \le \tau_0 \le T_0'\}$ функцию $D(\tau) = D(\tau_p, \tau_0) = \|E^p(\tau_p) - E^0(\tau_0)\|$. Это может быть опреде-

ленная каким-либо образом метрика в R^N, причем заметим, что $D(\tau) \geq 0 \quad \forall \tau \in T_p$. Введем в рассмотрение функционал

$$G_p(\tau,u) = \int_0^T D(\tau(t),u(t))dt, \qquad (1)$$

который будем называть мерой сходства образов, где $\tau(t)=(\tau_p,\tau_o)$ - фазовый вектор, $u(t)$ - вектор управления, связанные дифференциальным уравнением:

$$\frac{d\tau}{dt} = f(\tau, u). \qquad (2)$$

Тогда для определения оптимальной меры сходства эталона E^p и образа E^o необходимо решить задачу нахождения минимума функционала (I) при связях (2). При P эталонах соответственно необходимо решить P таких задач, а результатом распознавания будет эталон с номером:

$$p = \underset{p=\overline{1,P}}{\operatorname{argmin}} G_p.$$

Для численного решения задачи (I) сведем ее к задаче нелинейного программирования. Это можно сделать, например, следующим способом [2,с.195]. В пространстве (T_p,t) проведем гиперплоскости Σ_i: $t_i = i\Delta t$, где $i=\overline{1,2,\ldots,M}$, Δt - шаг интегрирования. Предполагая, что на интервале $[i\Delta t,(i+1)\Delta t]$ управляющая вектор-функция принимает постоянное значение, уравнение (2) можно заменить разностной схемой

$$\tau_{i+1}=\tau_i + \Delta t \cdot f(\tau_i,u_i). \qquad (3)$$

Соответственно интеграл (I) заменим суммой

$$G_p(\tau,u) = \Delta t \sum_{i=0}^{M-1} D(\tau_i,u_i). \qquad (4)$$

Таким образом, задачу вычисления меры сходства свели к следующей: определить векторы τ_i и u_i, доставляющие минимум (4) при связях (3) и условиях:

$$\tau_i \in T_p, \quad u_i \in V, \qquad (5)$$

где T_p, V - некоторые заданные множества.

Для определенности будем рассматривать задачу с фиксированным начальным условием и со свободным конечным, т.е.

$$\tau\big|_{t=0} = \tau_0 . \tag{6}$$

Тогда на основании равенства (3) можно последовательно исключить фазовые векторы τ_{i+1}:

$$\left.\begin{aligned}
&\tau_1 = \tau_0+\Delta t\cdot f(\tau_0,u_0)=F_1(u_0),\\
&\tau_2 = F_1(u_0)+\Delta t f(F_1(u_0),u_1) = F_2(u_0,u_1),\\
&\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\\
&\tau_k = F_{k-1}(u_0,u_1,\ldots,u_{k-2})+\Delta t f(F_{k-1},u_{k-1})=F_k(u_0,\ldots,u_{k-1}).
\end{aligned}\right\} \tag{7}$$

Функционал (4) становится только функцией векторов управления: $u_0,u_1,\ldots,u_{k-1}$, а задачу вычисления меры сходства можно записать в виде:

$$G = \min_{u_k} \sum_{k=0}^{K-1} I_k(u_0,u_1,\ldots,u_k), \tag{8}$$

где $I_k(u_0,u_1,\ldots,u_k) = \Delta t D(F_k(u_0,\ldots,u_{k-1}),u_k)$. Функции такого типа называются функциями с последовательным включением неизвестных. Заметим, что каждый член суммы (8) есть функция только первых K неизвестных.

Если, кроме начального условия (6), необходимо также удовлетворить некоторому конечному условию

$$\tau\big|_{t=K\Delta t} = \tau_{K0}, \tag{9}$$

то, не снижая общности, можно решать задачу со свободным концом, но следует видоизменить функционал

$$G = C\|\vec{a}\| + \Delta t \sum_{k=0}^{K-1} D(\tau_k,u_k), \tag{10}$$
$$a = (\tau_k-\tau_{k0}),$$

где $\vec{a}$ - некоторый вектор, равный величине рассогласования с конечным вектором, C - достаточно большая константа, а величина $C\|a\| = C\|\tau_k-\tau_{k0}\|$ - функция штрафа [3]. Тогда, чтобы получить решение, удовлетворяющее обоим условиям, необходимо в процессе итераций на основании некоторого предварительного решения добиться равенства нулю функции штрафа. Поэтому ниже будем рассматривать задачу (8) с фиксированным начальным условием.

Для нахождения минимума функционала (8) воспользуемся методом градиентного спуска [2, с.210]. Для этого сначала необходимо построить некоторое "диспетчерское" решение, желательно близкое к истинному решению. На основании неотрицательности функции $D(\tau)$ для получения приближенного решения перейдем от (8) к функционалу

$$G(\tau,u) = \Delta t \sum_{k=0}^{K-1} \min I_k(u_0,u_1,\ldots,u_k), \qquad (11)$$

при этом будем выполнять соотношение $\hat{G} \geq G$. Решение задачи (11) существенно проще, так как при этом она разбивается на k-1 задачу последовательной минимизации функций только одной переменной I_k. Тогда каждый шаг в методе градиентов сведется к расчету очередного приближения вектора u_k по формуле

$$u_k = \tilde{u}_k - \nu I_k = \tilde{u}_k - \nu \frac{\partial I_k}{\partial u_k},$$

где $\tilde{u}_k$ - предыдущее приближение, ν - шаг градиентного спуска, $\frac{\partial I_k}{\partial u_k}$ - вектор с компонентами

$$\frac{\partial I_k}{\partial u_{k1}}, \frac{\partial I_k}{\partial u_{k2}}. \qquad (12)$$

Получив таким способом диспетчерское решение, можно вернуться к функционалу (8) и продолжить уточнение этого результата, но в реальной задаче распознавания образов зачастую в этом не возникает необходимости, так как диспетчерское решение оказывается достаточно эффективной оценкой меры сходства образов.

2. Алгоритм распознавания дискретной речи

Реальная задача распознавания является дискретной, соотношения (2) не известны, функция $D(\tau)$ также задана в дискретном виде, поэтому непосредственно вычислять градиент из соотношения (12) нет возможности. Тогда можно воспользоваться следующей процедурой. Так как по условию (6) начальная точка лежит на оптимальной фазовой траектории $\tau(t)$, то на каждом шаге минимизации функций I_k необходимо варьировать вектор управления в допустимых пределах (5) и принимать в качестве оптимального вектор, доставляющий минимум функции $I_k(u_k)$. Подобную процедуру оптимизации следует отнести к методу возможных направлений [4], являющемуся об-

общением метода градиентного спуска с конечной длиной шага. Суть его заключается в следующем. Возможным называется направление, вдоль которого можно сделать шаг конечной длины и уменьшить при этом значение оптимизируемой функции, не выходя за пределы ее области определения.

Перепишем соотношения (7) в виде: $\tau_k = \tau_{k-1}(u_0,\ldots,u_{k-2}) + \lambda_{k-1}u_{k-1}$, и соответственно (II) можно переписать в виде:

$$\hat{G} = \sum_{k=0}^{K-1} \min D(\tau_k(u_0,\ldots,u_{k-1}),u_k), \qquad (13)$$

где u_k - вектор управления, который в данном случае будет вектором направления, λ_k - длина шага, причем $u_k \in V_k$ - множество возможных направлений, определяемое конкретными ограничениями на фазовые траектории $\tau_k(t)$, в частности обязательным является требование монотонности.

Определим функцию частичной меры сходства образов

$$\varepsilon_{k+1}(\tau_{k+1}) = \varepsilon_k + D(\tau_k,u_k).$$

Тогда решение (I3) сведется к последовательной минимизации на каждом шаге ε_{k+1}, а именно: на k+1 шаге выбираем одно из возможных направлений и вычисляем некоторое приближение функции $\hat{\varepsilon}_{k+1}$. Затем варьируем направление и выбираем доставляющее минимум u_k^*. Алгоритм вычисления оценки меры сходства можно представить в рекуррентном виде

$$u_k^* = \operatorname*{argmin}_{u_k \in V_k} D(\tau_k,u_k),$$
$$\varepsilon_{k+1}(\tau_{k+1}) = \varepsilon_k + \min_{u_k \in V_k} D(\tau_k,u_k) \qquad (14)$$

с начальным условием: $\varepsilon(\tau_1) = D(\tau_0,u_0)$, $u_0 = \text{const}$. Заметим, что так как эталоны имеют разную длину, то необходимо ввести весовую функцию для компенсации влияния числа членов суммы (I3) и переписать (I4) окончательно в виде:

$$\varepsilon_{k+1}(\tau_{k+1}) = \varepsilon_k + z(u_k^*)D(\tau_k,u_k^*),$$

где $\sum_{k=0}^{K-1} z(u_k^*) = \text{const}$ - сумма длин эталона и образа. Если необходимо удовлетворить конечному условию (9), то в окончательной мере сходства нужно учесть функцию штрафа (I0), определяя ее следующим

образом. При достижении фазовой траекторией границы области T_p в качестве штрафа использовать значение функции меры сходства, вычисляемое вдоль границы области T_p до точки τ_k.

Описанный алгоритм вычисления мер сходства образов реализован в системе распознавания речевых образов [I]. Он показал достаточно высокую эффективность как с точки зрения быстродействия, так и надежности распознавания.

Проведенное сравнение предложенного алгоритма с различными модификациями алгоритмов динамического программирования показало, что он позволяет получить надежность распознавания по крайней мере не ниже, чем последние, при требованиях вычислительных затрат в 10–50 раз меньших [5].

Тем не менее следует остановиться на некоторых особенностях реализации алгоритма в системе распознавания [I]. Исследования показали, что алгоритмы динамического программирования более устойчивы к случайным выбросам параметров речевого сигнала, вызванных нестабильностью работы устройств предпроцессорной обработки. Это следует из построения самого алгоритма градиентного спуска, основанного на исследовании локального поведения функции $D(\tau)$. Поэтому для устранения нежелательных случайных выбросов каждый речевой параметр в системе подвергается сглаживанию с помощью цифрового фильтра нижних частот I-го порядка: $e'_{k+1} = (1-\alpha)e'_k + \alpha e_k$, где $0 < \alpha < 1$ – параметр. При интервале дискретизации параметров в 7–10 мсек такое сглаживание практически не искажает структуры речевого сигнала, но объем информации при этом слишком велик. Поэтому после фильтрации сигнал подвергается нелинейной компрессии на основе следующего алгоритма. Пусть $E = (e'_1, e'_2, \ldots, e'_k, \ldots, e'_k)$ – описание речевого сигнала после фильтрации, где e'_k – векторы признаков, измеренные через равные интервалы времени, $D(e'_j\ e'_k)$ – метрика в пространстве признаков R^N. Пусть вектор e_j вторичного описания соответствует вектору с номером i первичного описания, тогда в качестве вектора e_{j+1} вторичного описания берется вектор e'_k первичного описания, для которого выполняется соотношение: $d(e'_i\ e'_k) \geq \Delta$, $k = i+1\ i+2, \ldots$, где Δ – некоторый порог, подбираемый экспериментально. Исследования показали, что компрессия, сокращающая исходное описание в 2–3 раза, не ухудшает, а, напротив, улучшает надежность распознавания. Это можно объяснить тем, что комп-

рессии в основном подвергаются только стационарные участки сигналов, что приводит к повышению "веса" участков, соответствующих согласным звукам и переходным.

Снижение вычислительных затрат на вычисление оценок сходства образов позволило реализовать в системе [I] два независимых алгоритма вычисления мер сходства: при заданном начальном и, наоборот, конечном условиях. Тогда при несовпадении результатов распознавания I и 2-го алгоритмов принимается решение о дополнительной проверке конкурирующих гипотез на основе какого-либо 3-го алгоритма разрешения конфликта либо происходит отказ от распознавания. Такая иерархическая процедура принятия решения значительно повышает устойчивость работы системы распознавания. В частности, при поступлении на вход системы образов, на которые система не обучена, с большой вероятностью происходит отказ от распознавания и требование повторить высказывание.

3. Алгоритм распознавания слитной речи

Пусть $\mathfrak{e} = \{E^P\}$ - множество эталонов и X - образ, представленные последовательностью векторов признаков, L_p - длина p-го эталона, Q - длина образа. Будем считать, что входной образ состоит из не более чем S слов. Тогда образу X можно сопоставить синтезированный из эталонов образ Γ_J, составленный из S слов:

$$\Gamma_J = E^{p_1} \oplus E^{p_2} \oplus \ldots \oplus E^{p_j} \oplus \ldots \oplus E^{p_J},$$

$$j \in \{1,S\}, \quad p_j \in \{1,P\},$$

где $\oplus$ - оператор "сцепления" эталонов слов. Кроме того, будем считать, что отсутствуют ограничения на порядок следования слов во фразах.

Для получения оценки сходства образа X и эталона Γ_J необходимо решить J задач, аналогичных (I4) со свободным вторым концом. При этом начальное условие для первой задачи запишется в виде:

$$\tau_0^{p_1} = (1,1), \; g(\tau_0^{p_1}) = D(e_1^{p_1}, X_1).$$

Тогда конечное условие I-й задачи определится в результате вычисления оценки меры сходства первого слова-эталона с соответствующим "отрезком" образа:

$$\tau_k^{p_1} = (L_{p_1}, q_{p_1}),\ g(\tau_k^{p_1}) = g(\tau_{k-1}^{p_1}) + z(u_{k-1}^*)D(\tau_k^{p_1}),$$

$$u_k^* = \underset{u_{k-1}\in V}{\operatorname{argmin}}\, D(\tau_{k-1}^{p_1}, u_{k-1}),$$

где L_{p_1} – длина I-го во фразе Γ_J эталона. Вычисленное конечное условие I-й задачи будем считать начальным условием для следующей и т.д. Тогда в обобщенном виде начальные и конечные условия можно записать:

$$\tau_0^{p_j} = (L_{p_1} + \ldots + L_{p_{j-1}}, q_{p_{j-1}}),$$

$$g(\tau_0^{p_j}) = g(\tau_k^{p_{j-1}}),$$

$$g(\tau_k^{p_j}) = g(\tau_{k-1}^{p_j}) + z(u_{k-1}^*)D(\tau_k^{p_j}).$$

Окончательное значение оценки меры сходства эталонной фразы Γ_J и образа X можно представить в виде суммы мер сходства эталонов слов, составляющих фразу, с соответствующими отрезками входного образа X:

$$G(\Gamma_J, X) = \frac{1}{\sum_{j=1}^{J} L_{p_j} + Q}\, g(\tau_k^{p_J}),$$

где L_{p_j} – длины эталонов и образа. Решением задачи распознавания будет фраза, состоящая из последовательности слов:

$$p_1^* p_2^* \ldots p_j^* \ldots p_J^* = \operatorname{argmin} \hat{G}(\Gamma_J, X),$$

$$p_j \in \{1, P\},\quad J = \overline{1,S}.$$

Алгоритм распознавания слитной речи реализован в экспериментальной системе распознавания фраз диспетчера управления воздушным движением. Объем словаря составлял 60 слов, из них 37 чисел от 0 до 19, 20, 30, ..., 100, ..., 900. Синтаксис был задан в виде графов; проход из истока графа в сток порождал одну из допустимых фраз, число которых составило около 600; максимальная длина фразы – 3 слова; максимальный коэффициент ветвления – 37. Время распознава-

ния на микро-ЭВМ составляет 2-4 секунды в зависимости от длины фразы, надежность распознавания порядка 80%. В настоящее время проводится совершенствование алгоритма согласования на стыках слов и алгоритмов создания эталонов с целью повышения надежности распознавания.

Литература

I. Система распознавания речи ДИС-332.03 /Горловский А.Л., Лендяшов Н.А.,Петров А.Н. и др.-В кн.: Автоматическое распознавание слуховых образов. Тезисы докл. и сообщ. АРСО-I3, Новосибирск, I984, ч.2, с. 95.

2. МОИСЕЕВ Н.Н. Элементы теории оптимальных систем. -М.:Наука, I975.

3. ПРОПОЙ А.И. Методы возможных направлений в задачах оптимального дискретного управления. -Автоматика и телемеханика, I967, № 2.

4. ЗОЙТЕНДЕЙК Г. Методы возможных направлений. -М.: ИЛ, I963.

5. ПОДМЕТКИН М.И., ТУРКИН В.Н. Комплекс программ для сравнительного анализа алгоритмов распознавания речи. -В кн.: Автоматическое распознавание слуховых образов. Тезисы докл. и сообщ. АРСО-I3, Новосибирск, I984, ч. 2, с. 97.

Поступила в ред.-изд.отд.
I2 декабря I984 года