О ПОНЯТИИ «ПОЛЕЗНАЯ ИНФОРМАЦИЯ»

М. М. БОНГАРД

(МОСКВА)

Теория информации появилась как теория передачи сообщений по каналам связи. Поэтому, естественно, она мало внимания уделяла вопросу ценности переданной информации для устройств, находящихся на приемном конце канала. Между тем имеется много вопросов, для решения которых количество переданных бит далеко не полностью характеризует информацию.

Существует также подход, при котором ценность информации должна характеризоваться изменением трудности некоторой задачи (см., например, [1], [2], [3]). При этом возникает целый ряд осложнений, связанных, во-первых, с тем, что «трудность» задачи может быть различной для разных наблюдателей, и, во-вторых, с тем, что изменение трудности зависит не только от пришедшего сообщения, но и от того, как наблюдатель к этому сообщению относится. Мы говорим про сообщение: «я этому не верю», «я это знал и раньше», «не понимаю» и т. п. Без формализации «истолкования сообщения наблюдателем», очевидно, нельзя говорить о пользе сообщения.

В настоящей работе будет рассматриваться система, которая для решения задачи ведет экспериментальную работу (метод проб и ошибок) и таким путем извлекает некоторые сведения, которых она вначале не имела. В качестве «трудности» задачи для этой системы принимается некоторая функция числа проб, необходимых для нахождения решения.

Кроме того, система может получать сведения о задаче через канал связи. Следствием этого является изменение порядка проб, а значит, и трудности задачи.

Этот подход был использован [4] при оценке «полезности» процесса обучения счетной машины для последующего узнавания некоторых объектов. За «трудность» задачи принималось просто среднее число проб, необходимых машине.

В большей части настоящей работы в качестве меры трудности используется логарифм среднего числа проб. Это вызвано тем, что при такой оценке получаются наиболее простые соотношения между пропускной способностью канала связи (или емкостью памяти) и максимальным сокращением трудности, которое можно получить, используя этот канал (или память).

Однако основные понятия введены, по существу, независимо от конкретной функции цены трудности и могут быть применены к довольно широкому классу функций цены. Одна из сторон этого вопроса рассматривается в § 8.

§ 1. Задача. Решающий алгоритм. Неопределенность

Пусть A есть множество элементов a_j. Имеется n подмножеств $b_1, b_2, \ldots, b_n$ множества A таких, что $b_1 \cup b_2 \cup \ldots \cup b_n = A$.

Пусть мы обладаем алгоритмом проверки утверждения: «$a_j \in b_i$»; назовем его *W-алгоритмом.*

Нас будет интересовать, какие a_j каким b_i принадлежат. Будем называть A *— массовой задачей*, a_j *— частным случаем задачи* A и b_k *— ответом этой частной задачи*, если $a_j \in b_k$.

Будем считать, что имеется некоторый способ выбора элементов a_j из множества A такой, что для каждого элемента существует вероятность $p(a_j)$ быть выбранным. В соответствии с этим будем говорить: «нам нужно решить задачу A», если с вероятностью $p(a_1)$ нам придется решать частную задачу a_1, с вероятностью $p(a_2)$ — задачу a_2 и т. д.

В дальнейшем мы рассматриваем следующий класс алгоритмов, которые называются *решающими.* Выбираем с помощью жребия с некоторым распределением вероятностей одно из множеств b_i. Затем подставляем в W-алгоритм решаемую частную задачу a_j и выбранное с помощью жребия b_i. Если ответ — «истина», то задача решена, если — «ложь», то снова бросаем жребий, и т. д. При этом распределение вероятностей быть выбранными для разных b_i может оставаться постоянным, а может изменяться от шага к шагу.

Различные решающие алгоритмы отличаются друг от друга начальными распределениями вероятностей и законами их изменения.

Если заданы частная задача a_j и решающий алгоритм, то число применений W-алгоритма (проб), приводящее к отысканию ответа, есть случайная величина $K(a_j)$. Назовем *неопределенностью* $N(a_j)$ частной задачи a_j для данного решающего алгоритма логарифм математического ожидания $\overline{K}(a_j)$ числа проб, т. е.

$$N(a_j) = \log \overline{K}(a_j). \tag{1}$$

Неопределенностью задачи A для этого решающего алгоритма будем называть среднюю неопределенность частных задач a_j

$$N(A) = \sum_j p(a_j) N(a_j) = \sum_j p(a_j) \log \overline{K}(a_j). \tag{2}$$

§ 2. Связь между неопределенностью и энтропией

В §§ 2 — 5 мы будем всюду рассматривать, не оговаривая этого каждый раз, такие задачи A, все частные задачи которых имеют только по одному ответу *) и решающие алгоритмы с постоянным распределением вероятностей выбора b_i.

Найдем неопределенность A в случае, если частная задача из A с вероятностью p_i **) имеет ответ b_i, а решающий алгоритм пробует b_i с вероятностью q_i (где $\sum_{i=1}^{n} q_i = 1$).

Если решается частная задача с ответом b_i, то среднее число проб равно $1/q_i$ ***). Обозначим неопределенность в этом случае через $N_i(A)$. Тогда

$$N_i(A) = -\log q_i.$$

*) Это эквивалентно утверждению, что $b_i \cap b_k = 0$ при $i \neq k$.

**) p_i есть сумма вероятностей $p(a_j)$ для всех частных задач, имеющих ответ b_i.

***) При этом мы используем то, что b_i — единственный ответ. Ср. § 6.

Вероятность этой ситуации равна p_i. Следовательно,

$$N(A) = N(p/q) = -\sum_{i=1}^{n} p_i \log q_i. \tag{3}$$

Легко показать (см., например, [2], [5]), что если p_i фиксированы, то неопределенность задачи минимальна тогда, когда $q_i = p_i$. В этом случае неопределенность задачи равна энтропии распределения вероятностей $p_1, p_2, \ldots, p_i, \ldots, p_n$, которую обозначим через $H(p)$. В общем случае

$$N(p/q) \geqslant H(p). \tag{4}$$

Таким образом, *если* при построении решающего алгоритма *мы знаем* вероятности p_i, то лучшее, что можно сделать, — это выбрать $q_i = p_i$ (лучшее в смысле минимизации неопределенности задачи для решающего алгоритма).

Допустим, имеется задача с распределением вероятностей $p_1, p_2, \ldots, p_n$. Мы же, по какой-то причине, полагаем, что распределение вероятностей $p'_1, p'_2, \ldots, p'_n$, причем, вообще говоря, $p'_i \neq p_i$. Стремясь построить наилучший решающий алгоритм, мы сделаем $q_i = p'_i$ $(i = 1, 2, \ldots, n)$. Неопределенность при этом будет $N(A) = N(p/p')$.

Мы видим, что выражение (3) можно рассматривать также как неопределенность задачи с распределением вероятностей ответа p для наблюдателя, исходящего из гипотезы, что это распределение равно q. Поэтому мы будем называть распределение вероятностей q-гипотезой наблюдателя.

С этой точки зрения энтропия $H(p)$ является частным случаем неопределенности. Энтропия — неопределенность для алгоритма, знающего и полностью использующего распределение вероятностей ответов задачи. Естественно считать, что энтропия — это неопределенность, связанная с самой задачей, а величина $N(p/q) - H(p)$ характеризует степень «незнания задачи» данным решающим алгоритмом.

Интуитивно очевидно, что, вообще говоря, незнание по своей природе может быть двоякого рода. Мы можем не все знать о задаче, и мы можем «знать» то, чего нет на самом деле (заблуждаться). Для того чтобы формально разделить эти случаи, рассмотрим ситуацию, когда при построении решающего алгоритма мы знаем не распределение вероятностей p, а лишь некоторые ограничения типа:

$$\left.\begin{aligned} f_1(p_1, p_2, \ldots, p_n) &\geqslant 0, \\ \ldots\ldots\ldots\ldots\ldots & \\ f_k(p_1, p_2, \ldots, p_n) &\geqslant 0. \end{aligned}\right\} \tag{5}$$

К системе ограничений, конечно, всегда добавляются соотношения

$$p_i \geqslant 0 \quad \text{и} \quad \sum_{i=1}^{n} p_i = 1. \tag{5'}$$

Будем считать, что система ограничений такова, что существует, по крайней мере, одно распределение p, удовлетворяющее всем неравенствам.

Введем следующие определения.

Распределение вероятностей q^0 называется *оптимальной гипотезой* относительно ограничений (5), если

$$\max_p N(p/q^0) \leqslant \max_p N(p/q), \tag{6}$$

где $\max_p$ есть точная верхняя граница, взятая по всем распределениям $\boldsymbol{p}$, удовлетворяющим ограничениям (5), а $\boldsymbol{q}$ — произвольная гипотеза.

Назовем распределение вероятностей $\tilde{\boldsymbol{q}}$ *гипотезой, содержащей только истину относительно ограничений* (5), если

$$\max_p N(\boldsymbol{p}/\tilde{\boldsymbol{q}}) \leqslant H(\tilde{\boldsymbol{q}}). \tag{7}$$

Гипотезы, не удовлетворяющие соотношению (7), мы будем называть *содержащими не только истину*. Заметим, что для любой системы ограничений существует, по крайней мере, одна содержащая только истину гипотеза $\tilde{q}_i = \frac{1}{n}$ $(i = 1, 2, \ldots, n)$. Действительно, в этом случае

$$N(\boldsymbol{p}/\tilde{\boldsymbol{q}}) = \log n = H(\tilde{\boldsymbol{q}}).$$

Назовем $\tilde{\boldsymbol{q}}^0$ *сильнейшей содержащей только истину гипотезой*, если $\tilde{\boldsymbol{q}}^0$ — гипотеза, содержащая только истину, и имеет место соотношение

$$H(\tilde{\boldsymbol{q}}^0) \leqslant H(\tilde{\boldsymbol{q}}), \tag{8}$$

где $\tilde{\boldsymbol{q}}$ — произвольная, содержащая только истину гипотеза.

Можно показать *), что множество всех гипотез, содержащих только истину, является замкнутым, и так как $H(\tilde{\boldsymbol{q}}) \geqslant 0$, то $H(\tilde{\boldsymbol{q}})$ достигает

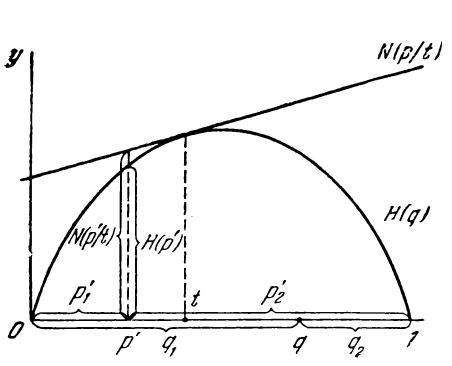

Рис. 1.

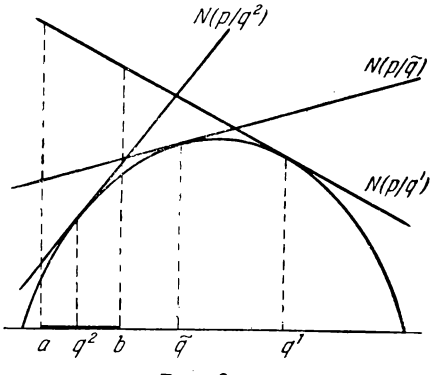

Рис. 2.

на этом множестве точной нижней границы, а следовательно, сильнейшая содержащая только истину гипотеза всегда существует.

Для дальнейшего нам будет удобно обратиться к геометрической интерпретации понятий: «распределение вероятностей ответов задачи», «гипотеза», «энтропия» и «неопределенность».

Рассмотрим случай $n = 2$. Благодаря связи $q_1 + q_2 = 1$ многообразие всех возможных гипотез является одномерным. Будем изображать гипотезу точкой $\boldsymbol{q}$ на отрезке [0; 1] оси абсцисс (рис. 1). На этом же отрезке находятся точки $\boldsymbol{p}$. Для отыскания неопределенности задачи, характеризуемой $\boldsymbol{p}'$, при гипотезе $\boldsymbol{q} = \boldsymbol{t}$ строим кривую $y = H(\boldsymbol{q})$ и проводим прямую, касательную к ней в точке, соответствующей $\boldsymbol{t}$. Касательная будет графиком функции $y = N(\boldsymbol{p}/\boldsymbol{t})$.

В самом деле, $N(\boldsymbol{p}/\boldsymbol{t}) = -p_1 \log t_1 - p_2 \log t_2$ есть линейная функция от $\boldsymbol{p}$. $N(\boldsymbol{t}/\boldsymbol{t}) = H(\boldsymbol{t})$ и $N(\boldsymbol{p}'/\boldsymbol{t}) > H(\boldsymbol{p}')$ при $\boldsymbol{p}' \neq \boldsymbol{t}$. Таким образом, $y = N(\boldsymbol{p}/\boldsymbol{t})$ должно быть прямой, имеющей с $y = H(\boldsymbol{q})$ общую точку при $\boldsymbol{p} = \boldsymbol{t}$ и лежащей выше $y = H(\boldsymbol{q})$ при $\boldsymbol{p} \neq \boldsymbol{t}$. Так как, кроме того, $y = H(\boldsymbol{q})$ выпукла, непрерывна и имеет непрерывную производную всюду, кроме точек $q_1 = 0$ и $q_1 = 1$, то $y = N(\boldsymbol{p}/\boldsymbol{t})$ — касательная к $y = H(\boldsymbol{q})$.

*) Для краткости доказательство мы не приводим.

Пусть ограничения, наложенные на p, например, таковы, что p может находиться только на отрезке ab (рис. 2). Тогда из изображенных на рисунке гипотез гипотеза $\tilde{q}$ содержит только истину относительно этих ограничений, а q^1 и q^2 — гипотезы, содержащие не только истину, так как

$$\max_p N(p/q^1) > H(q^1) \text{ и } \max_p N(p/q^2) > H(q^2).$$

Очевидно, что в этом примере $\tilde{q}^0$ совпадает с точкой b. Оптимальная гипотеза q^0 также совпадает с b, так как

$$\max_p N(p/b) = H(b), \quad \text{а} \quad \max_p N(p/q) \geqslant N(b/q) > H(b)$$

при $q \neq b$.

Если n произвольно, то p и q являются точками $(n-1)$-мерного тетраэдра. Обозначим $(n-1)$-мерную гиперплоскость, содержащую этот тетраэдр, через K. Обозначим через $\bar{L}$ множество всех точек тетраэдра, удовлетворяющих ограничениям (5), а через L — замкнутое множество, содержащее $\bar{L}$ и его предельные точки. В этом случае $y = H(q)$ есть

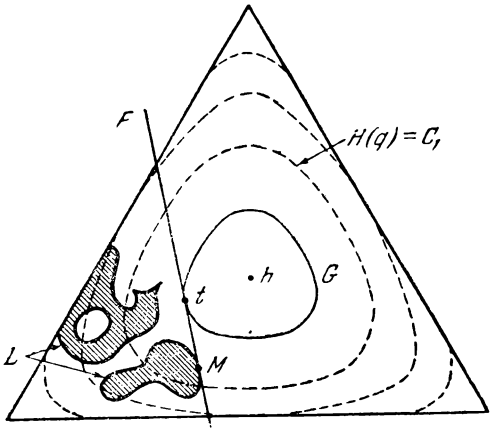

Рис. 3.

$(n-1)$-мерная выпуклая поверхность. Проведем $(n-1)$-мерную гиперплоскость, касательную к поверхности $y = H(q)$ в точке, соответствующей $q = t$. Уравнение этой гиперплоскости $y = N(p/t)$. Посмотрим, какими свойствами она обладает. Образуем $(n-2)$-мерное пересечение гиперплоскостей $y = N(p/t)$ и $y = H(t) = \text{const}$. Спроектируем полученное пересечение на K, эта проекция F также является $(n-2)$-мерной плоскостью. Если $p \in F$, то $N(p/t) = H(t)$. F делит K на две части. Для точек p, находящихся по одну сторону F, имеет место соотношение $N(p/t) < H(t)$, для находящихся по другую сторону — соотношение $N(p/t) > H(t)$. Точка h с координатами $h_i = \frac{1}{n}$ $(i = 1, 2, \ldots, n)$ всегда попадает в область больших неопределенностей. Действительно, $N(h/t) \geqslant H(h) \geqslant H(t)$. При $t \neq h$ имеет место строгое неравенство. При $t = h$ плоскость $y = N(p/t)$ совпадает с $y = H(t)$ и F заполняет весь тетраэдр.

Спроектируем теперь на K область, образованную пересечением поверхности $y = H(q)$ и плоскости $y = H(t)$. Обозначим эту проекцию через G. Очевидно, $t \in F$, $t \in G$ и F касается G в точке t.

Пересечение L и F обозначим через M.

На рис. 3 изображена плоскость K для случая $n = 3$. Для того чтобы узнать, является ли t гипотезой, содержащей только истину

относительно ограничений (5) (или, что то же самое, относительно области L), нужно провести через t поверхность постоянной энтропии G и построить касательную к ней в точке t гиперплоскость F. Если F отделяет L от h*), то t есть гипотеза, содержащая только истину**).

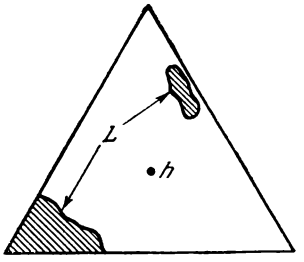

Рис. 4.

Отсюда следствие: если область L такова, что не существует $(n-2)$-мерной гиперплоскости, отделяющей ее от h, то единственной содержащей только истину гипотезой является $\tilde{q}=h$. Пример такого случая показан на рис. 4.

Теорема. *Для любой системы ограничений, наложенных на* p, *существует единственная оптимальная гипотеза* q^0. *Оптимальная гипотеза* q^0 *совпадает с сильнейшей содержащей только истину гипотезой* $\tilde{q}^0$. *При этом*

$$\max_p N(p/q^0) = H(q^0).$$

Докажем сначала следующую лемму.

Лемма. Если $\tilde{q}^0$ — *сильнейшая содержащая только истину гипотеза, то найдутся такие точки* $m^1; m^2; \ldots m^j \ldots; m^s$, $m^j \in M$, $s \leqslant n-1$, *и такие числа* $\alpha^j > 0$, *что* $\sum_{j=1}^{s} \alpha^j = 1$ *и* $\sum_{j=1}^{s} \alpha^j m^j = \tilde{q}^0$.

Пусть B — выпуклая оболочка, натянутая на область L. Рассмотрим соотношение между B и поверхностью G, проходящей через $\tilde{q}^0$. В принципе мыслимы четыре случая: 1) B и G пересекаются, 2) B и G не имеют общих точек, 3) B и G касаются, но не в точке $\tilde{q}^0$, 4) B и G касаются в точке $\tilde{q}^0$.

Первый случай не может иметь места, так как при этом не существует гиперплоскость, отделяющая B (а значит, и L) от G.

Реализация второго случая противоречила бы предположению, что $\tilde{q}^0$ есть *сильнейшая* содержащая только истину гипотеза. Действительно, мы могли бы построить другую поверхность G', охватывающую G и не пересекающуюся с B. На G' всегда найдется точка t такая, что гиперплоскость, касательная в этой точке к G' отделяет G' от B. Значит, t было бы гипотезой, содержащей только истину. Но $H(t) < H(\tilde{q}^0)$, что и доказывает противоречивость предположений.

Третий случай, как и первый, не совместим с предположением, что $\tilde{q}^0$ содержит только истину. В самом деле, G есть выпуклая поверхность, не имеющая плоских частей. Поэтому гиперплоскости, касательные к ней в двух разных точках, не могут совпадать. Так как B и G касаются, то единственная разделяющая их гиперплоскость касается G в этой же точке, а значит, касательная к G в точке $\tilde{q}^0$ гиперплоскость не отделяет B от G.

Итак, остается только четвертая возможность. Это значит, что $\tilde{q}^0$ обязательно принадлежит B. Так как B есть выпуклая оболочка области L, то $\tilde{q}^0$ является центром тяжести нескольких точек $m^1, m^2, \ldots, m^j, \ldots, m^s$ области L, которым приписаны положительные веса.

*) L может иметь точки на F.
**) При этом F отделяет L от G.

Последнее утверждение и означает, что найдутся такие числа α^j и точки $m^j \in L$, что

$$\sum_{j=1}^{s} \alpha^j = 1, \tag{9}$$

$$\sum_{j=1}^{s} \alpha^j m^j = \tilde{q}^0, \tag{10}$$

причем

$$\alpha^j > 0 \qquad (j = 1, 2, \ldots, s). \tag{11}$$

Так как $\tilde{q}^0$ принадлежит гиперплоскости F, а L расположена по одну сторону от F, то $m^j \in F$ и, следовательно, $m^j \in M$.

Лемма доказана.

Следствие. *Если* $t = \tilde{q}^0$, *то* M *не пусто.*

Отсюда

$$\max_p N(p/\tilde{q}^0) = H(\tilde{q}^0). \tag{12}$$

Переходим к доказательству теоремы. Из формулы (10) следует:

$$N(\tilde{q}^0/q) = \sum_{j=1}^{s} \alpha^j N(m^j/q). \tag{13}$$

Учитывая (9) и (11), получаем:

$$N(\tilde{q}^0/q) \leqslant \max_j N(m^j/q),$$

где $\max_j$ есть максимум, взятый по s точкам m^j. Так как $m^j \in L$, то

$$N(\tilde{q}^0/q) \leqslant \max_p N(p/q). \tag{14}$$

В случае $q \neq \tilde{q}^0$

$$H(\tilde{q}^0) < N(\tilde{q}^0/q). \tag{15}$$

Сопоставив (12), (15) и (14), получаем:

$$\max_p N(p/\tilde{q}^0) < \max_p N(p/q), \tag{16}$$

это и означает, что $\tilde{q}^0$ есть оптимальная гипотеза, и притом единственная.

Следствия. а) *Из соотношения* (12) *вытекает*:

$$\max_p N(p/q^0) = H(q^0). \tag{17}$$

б) *Из единственности оптимальной гипотезы следует единственность сильнейшей содержащей только истину гипотезы.*

в) *Если область* L *такова, что не существует* $(n-2)$*-мерной гиперплоскости, отделяющей ее от точки* h, *то оптимальной гипотезой является* $q^0 = h$. Например, если на p не наложено никаких ограничений, кроме $\sum p_i = 1$, то оптимальной будет гипотеза $q_i^0 = \frac{1}{n}$.

Следствие в) поясняет, в каком смысле целесообразно пользоваться распространенным приемом: «поскольку мы не имеем сведений о вероятностях некоторых событий, будем считать их равновероятными».

Объясним теперь, почему гипотеза, удовлетворяющая соотношению (7), названа «содержащей только истину». Пусть неравенства (5) определяют некоторую область L (рис. 5, *а*), внутри которой *в действительности* может находиться p. Допустим, нам сообщили *не все* неравенства. Другими словами, нам сообщили «только истину, но не всю истину». Сообщенные нам ограничения определяют область L'

Очевидно, $L' \supset L$. Если наша задача состоит в том, чтобы найти оптимальную гипотезу, то, при имеющихся в нашем распоряжении сведениях, мы выберем q^0-гипотезу, оптимальную относительно области L'. Относительно области L она не обязательно будет оптимальной, но всегда будет *содержащей только истину*.

Рассмотрим другой случай. Нам сообщили некоторые сведения, определяющие область L' (рис. 5, *б*). Исходя из них, мы выбрали

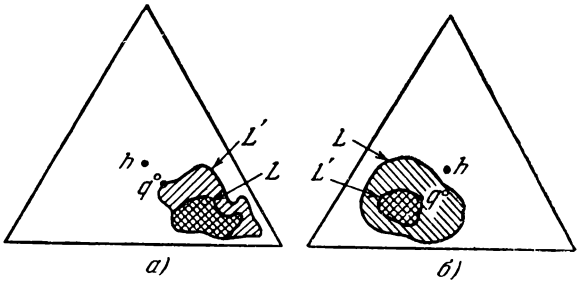

Рис. 5.

оптимальную гипотезу q^0. Если *в действительности* положение p ограничено не областью L', а L, то это значит, что нам сообщили *ложные* ограничения на положение p. И действительно, в случае, изображенном на рис. 5, *б*, q^0 содержит не только истину относительно области L.

Таким образом:

1) если при построении оптимальной гипотезы мы пользовались истиной (но не обязательно всей), то получим гипотезу, содержащую только истину;

2) если при построении оптимальной гипотезы мы получили гипотезу, содержащую не только истину, то значит сведения, которыми мы пользовались, содержали ложные ограничения области, где может находиться p.

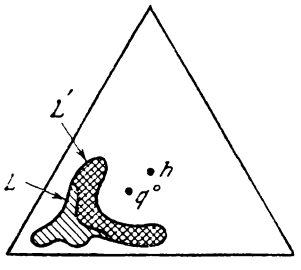

Рис. 6.

Разумеется, обратные соотношения не имеют места. Например, ложные ограничения могут и не привести к гипотезе, содержащей не только истину. Такой случай приведен на рис. 6. Очевидно, здесь дело в том, что не все свойства области L существенны для построения оптимальной гипотезы. Ложные сведения коснулись именно этих несущественных для нашей задачи свойств.

Равенство (17) говорит о том, что если мы пользуемся гипотезой, оптимальной при *реально существующих ограничениях*, то максимум возможной неопределенности равен энтропии гипотезы. Этим, по существу, определяются границы применимости энтропии как меры неопределенности. В общем случае, когда гипотеза может быть неоптимальной, неопределенность не выражается через энтропии распределений p или q.

* *
*

Не следует думать, что для введения понятия «неопределенность» обязательно нужно рассматривать решающий алгоритм, использующий случайный процесс («жребий»). Пусть мы обладаем W'-алгоритмом, дающим возможность про любое подмножество множества A сказать, принадлежит ли ему a_j. Припишем каждому b_i вес q_i, $\sum_{i=1}^{n} q_i = 1$. Сгруппи-

руем все b_i в два подмножества так, чтобы сумма весов b_i, вошедших в каждое подмножество, была по возможности близка к $1/2$. С помощью W'-алгоритма выясняем, в какой половине находится ответ. Разбиваем это подмножество на две части с суммами весов $\approx 1/4$ и т. д., пока не дойдем до подмножества, содержащего лишь одно b_i. Если ответом является b_k, которому мы приписали вес q_k, то нам понадобится $\approx -\log_2 q_k$ проверок с помощью W'-алгоритма. Отсюда, если ответ имеет распределение вероятностей $\boldsymbol{p}$, а мы приписали b_i вес q_i, то среднее число применений W'-алгоритма будет $\approx -\sum_{i=1}^{n} p_i \log_2 q_i$.

Все соотношения в предыдущем абзаце приблизительны из-за невозможности при произвольных q_i создавать подмножества с суммарным весом, в точности равным $1/2^k$. Именно поэтому для основного определения «неопределенности» мы воспользовались W-алгоритмом, хотя W'-алгоритм и имеет преимущество более «естественной» платы за применение.

§ 3. Декодирование сигнала. Полезная информация

Рассмотрим систему, состоящую из решающего алгоритма, канала связи и алгоритма, соединяющего выходной конец канала с решающим алгоритмом. Этот дополнительный алгоритм мы будем называть декодирующим. Каждому значению приходящего сигнала и распределению вероятностей $\boldsymbol{q}$ декодирующий алгоритм ставит в соответствие некоторое новое распределение $\boldsymbol{q}'$ и заменяет в решающем алгоритме q_i на q_i' *). Таким образом, пришедшее сообщение изменяет неопределенность задачи. Пусть до прихода сообщения задача имела для данного решающего алгоритма неопределенность N_0, а после прихода сообщения N_1. Мы будем говорить, что по каналу передана полезная информация

$$I_{\text{п}} = N_0 - N_1. \tag{18}$$

Из определения следует, что не имеет смысла говорить о полезной информации, содержащейся в сигнале, если не указаны: задача, которая решается, начальное состояние решающего алгоритма и свойства декодирующего алгоритма.

Учитывая сказанное в предыдущем параграфе, в случаях, когда до прихода сообщения на распределение $\boldsymbol{p}$ не наложено никаких ограничений (кроме вытекающих из самого смысла $\boldsymbol{p}$), мы будем считать, что решающий алгоритм пользуется гипотезой $q_i = \frac{1}{n}$ $(i = 1, 2, \ldots, n)$, и от этого уровня отсчитывать полезную информацию.

Изменение неопределенности задачи под влиянием пришедшего сигнала можно интерпретировать как процесс запасания полезной информации в виде распределения вероятностей $\boldsymbol{q}$. Если за нулевой уровень принять запас информации при $q_i = \frac{1}{n}$ $(i = 1, 2, \ldots, n)$, то запас полезной информации, содержащийся в гипотезе $\boldsymbol{q}$ относительно задачи с распределением вероятностей ответа $\boldsymbol{p}$, дается соотношением

$$I_{\text{п}} = \log n - N(\boldsymbol{p}/\boldsymbol{q}).$$

Точность, с которой можно вычислить $I_{\text{п}}$, зависит, в частности, от точности, с которой известны q_i. Однако сама величина $I_{\text{п}}$ не зависит

*) В терминах теории игр декодирующий алгоритм — это стратегия, являющаяся функцией приходящего сигнала.

от точности q_i. Этим «полезная информация» отличается от «просто информации» I, которая может храниться в коде набора $q_1, \ldots, q_n$. Действительно, I определяется логарифмом числа возможных различных состояний набора $q_1, \ldots, q_n$, т. е. прямо связана с точностью, с которой могут быть измерены q_i.

Разумеется, $I_{\text{п}} \leqslant I$. В самом деле, даже если каждая вероятность q_i может принимать только два значения, то число R различных наборов q не меньше n *).

Так как $I_{\text{п}} \leqslant \log n$, получаем:

$$I_{\text{п}} \leqslant \log n \leqslant \log R = I.$$

Приведем пример, иллюстрирующий определения. Некто A хочет застать в учреждении, открытом с 10 до 18 часов, сотрудника B, о котором известно, что он бывает там по два часа ежедневно. Желая уточнить эти сведения, A обратился к знакомому, работающему в том же учреждении. Знакомый ответил, что B бывает на работе после 14 часов в 2 раза чаще, чем до 14. После этого A позвонил секретарю в учреждение. В ответ на свой вопрос он услышал: «... на будущий месяц расписание еще не составлено, но товарищ B всегда принимает 5 раз в неделю с 12 до 14 и 1 раз с 14 до 16. Ой, простите, я ошиблась. Один раз в неделю с 12 до 14 и 5 раз с 14 до 16». Последний ответ соответствовал действительности.

Какую полезную информацию получил по интересующему его вопросу A:

1) из ответа знакомого?

2) из ответа секретаря?

3) получил бы из ответа секретаря, если бы она не исправила ошибки?

Для удобства все вероятности сведены в таблицу 1.

Таблица 1

	10—12	12—14	14—16	16—18
Начальная гипотеза . .	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$	$\frac{1}{4}$
Гипотеза после сообщения знакомого	$\frac{1}{6}$	$\frac{1}{6}$	$\frac{1}{3}$	$\frac{1}{3}$
Гипотеза после ответа секретаря	0	$\frac{1}{6}$	$\frac{5}{6}$	0
Гипотеза после ошибочного ответа секретаря	0	$\frac{5}{6}$	$\frac{1}{6}$	0
Истинное распределение вероятностей	0	$\frac{1}{6}$	$\frac{5}{6}$	0

*) Может показаться, что этих значений не меньше чем 2^n, однако связь $\sum_{i=1}^{n} q_i = 1$ сильно уменьшает число возможных векторов. Например, если q_i может быть только 1 или 0, то имеется лишь n разных векторов:

1, 0, 0, ..., 0; 0, 1, 0, ..., 0; ...; 0, 0, 0, ..., 0, 1.

Будем выражать неопределенность и полезную информацию в битах. Начальная неопределенность:

$$N_0 = -\frac{1}{6}\log_2\frac{1}{4} - \frac{5}{6}\log_2\frac{1}{4} = \log_2 4 = 2.$$

Неопределенность после ответа знакомого:

$$N_1 = -\frac{1}{6}\log_2\frac{1}{6} - \frac{5}{6}\log_2\frac{1}{3} \approx 1{,}75.$$

Неопределенность после ответа секретаря:

$$N_2 = -\frac{1}{6}\log_2\frac{1}{6} - \frac{5}{6}\log_2\frac{5}{6} \approx 0{,}65.$$

Если бы секретарь не исправила ошибки, была бы неопределенность:

$$N_3 = -\frac{1}{6}\log_2\frac{5}{6} - \frac{5}{6}\log_2\frac{1}{6} \approx 2{,}20.$$

Таким образом, полезная информация, полученная от знакомого:

$$I_{п_1} = N_0 - N_1 \approx 0{,}25.$$

Ответ секретаря содержал полезную информацию:

$$I_{п_2} = N_1 - N_2 \approx 1{,}1.$$

Если бы тот же самый разговор с секретарем произошел до ответа знакомого, то он содержал бы $N_0 - N_2 = 1{,}35$ бита полезной информации.

Ошибочный же ответ нес полезную информацию $I_{п_3} = N_1 - N_3 \approx \approx -0{,}45$. Другими словами, ошибочный ответ содержал 0,45 бита дезинформации.

При рассмотрении этого примера мы молчаливо подразумевали, что алгоритм декодирования сообщения сводится к переносу сообщенных вероятностей прямо в решающий алгоритм («полное доверие»). Разумеется, при другом декодирующем алгоритме полезная информация, содержащаяся в рассмотренных сообщениях, была бы другой. Например, со стороны A было бы естественно, получив ошибочное сообщение секретаря, резко противоречащее сообщению знакомого, не поверить ему, а верному сообщению поверить не полностью, во-первых, из-за некоторого расхождения с сообщением знакомого, во-вторых, из-за предварительной ошибки секретаря.

Следует отметить, что величина полезной информации зависит также от кода, которым передается сообщение. «Те же» сообщения могли быть иначе закодированы, например произнесены на незнакомом A языке. В этом случае декодирующее устройство вообще не перераспределило бы q_i в решающем алгоритме (A не понял бы даже, что сигнал имеет отношение к интересующему его вопросу). Полезная информация была бы равна нулю.

Во всех руководствах по теории информации настойчиво подчеркивается, что если в сообщении, закодированном с помощью двузначного алфавита, заменить «нули» на «единицы» и обратно, то количество информации не изменится. К полезной информации это, естественно, не относится.

§ 4. Пропускная способность канала связи для полезной информации. Запас полезной информации в декодирующем алгоритме

Рассмотрим теперь соотношение между запасом полезной информации в решающем алгоритме (после прихода сообщения) и пропускной способностью канала связи. Можно было бы ожидать, что прирост полезной информации не может быть больше информации, переданной по каналу.

Однако это не так. Существуют системы, для которых полезная информация, поступившая в решающий алгоритм, может быть много больше логарифма числа сигналов, различимых системой.

Прежде чем привести пример системы, обладающей таким свойством, уточним, что мы называем «неразличимыми сигналами». Два сигнала C_1 и C_2 называются *неразличимыми* для системы, если после прихода как C_1, так и C_2 система перейдет в одно и то же состояние, иначе говоря, решающий алгоритм будет пользоваться одной и той же гипотезой $q'_1, q'_2, \ldots, q'_n$. Задавшись определенной точностью измерения q_i, мы можем про любые два сигнала сказать, различимы они или нет. Очевидно, что реальная система может иметь лишь конечное число состояний. Пусть число различных наборов $q_1, \ldots, q_n$ равно t. Разобьем множество всех сигналов на классы $C_1, C_2, \ldots, C_m$ неразличимых между собой сигналов. Очевидно, $m \leqslant t$ *).

Заметим, что при таком подходе нам не нужно рассматривать последовательности сигналов, учитывать их длину, число символов и т. д. *Всякий* сигнал принадлежит какому-то C_k, поэтому любая последовательность сигналов (декодируемая один раз, как единый сигнал) неотличима от одного из сигналов нашего набора. Таким образом, совокупность канала связи и декодирующего устройства имеет ограниченную пропускную способность не в единицу времени и не на символ, а на сообщение в целом.

Свойства декодирующего устройства при фиксированном начальном распределении q исчерпываются разбиением сигналов на классы и таблицей 2, определяющей распределение q'_i после прихода сигнала из класса C_k.

Таблица 2

	C_1	C_2	...	C_k	...	C_m
q'_1	$d_{1,1}$	$d_{1,2}$	...	$d_{1,k}$	...	$d_{1,m}$
q'_2	$d_{2,1}$	$d_{2,2}$	...	$d_{2,k}$	...	$d_{2,m}$
...	...	...	...	...	...	...
q'_i	$d_{i,1}$	$d_{i,2}$	...	$d_{i,k}$	...	$d_{i,m}$
...	...	...	...	...	...	...
q'_n	$d_{n,1}$	$d_{n,2}$	...	$d_{n,k}$	...	$d_{n,m}$

Здесь $d_{1,k}, d_{2,k}, \ldots, d_{n,k}$ — значения $q'_1, q'_2, \ldots, q'_n$ после прихода сигнала из класса C_k, поэтому $d_{i,k} \geqslant 0$ и $\sum_{i=1}^{n} d_{i,k} = 1$ **). Всякую таблицу, составленную из $d_{i,k}$, обладающую такими свойствами, можно считать декодирующим устройством.

Вернемся к соотношению между разнообразием различимых сигналов и запасом полезной информации в решающем алгоритме после прихода сообщения. Рассмотрим пример.

В 1943 г. во французский город с населением около 50 000 человек был послан из Англии человек для связи с движением Сопротивления. Перед отъездом он получил запечатанный пакет и следующую инструк-

*) Сигналы, неразличимые для системы, находящейся в одном состоянии, могут оказаться различимыми для системы, находящейся в другом состоянии. Однако нам в дальнейшем достаточно рассматривать лишь классы сигналов, не различимых системой, находящейся в данном состоянии.

**) Кроме того, любые два столбца различны.

цию: «Постарайтесь сами найти людей, через которых можно связаться с партизанами. Если это не удастся, то можете запросить меня шифровкой по радио. Если я отвечу словом «вперед», то вскройте пакет — в нем адрес надежного человека. Ни в коем случае не вскрывайте пакета до получения сигнала «вперед», так как явка может быть открыта немцами. Никаким другим указаниям и советам по радио не верьте, ибо я не вполне уверен в некоторых своих сотрудниках».

Каково в данной ситуации разнообразие различимых сигналов и какую полезную информацию содержит сигнал «вперед»? Инструкция разбила множество сообщений на два класса. Первый содержит единственный элемент — сигнал «вперед». После прихода этого сигнала вероятности обращения ко всем жителям города, кроме одного (указанного в пакете), становятся равными нулю, а вероятность обращения к этому человеку становится единицей. Второй класс содержит все остальные сигналы. Каждый из них не изменяет вероятностей обращения к различным людям, а следовательно, сигналы, входящие во второй класс, неразличимы между собой. Итак, набор различимых сигналов содержит только два сигнала. В то же время запас полезной информации в решающем алгоритме после сигнала «вперед» увеличивается гораздо больше, чем на один бит. Действительно, неопределенность после сигнала — нуль, а до сигнала ≈ 10—12 бит (учитывая, что не надо проверять детей и что пригодных людей, вероятно, несколько).

При этом существенно то, что большая полезная информация, содержащаяся в сигнале «вперед», никак не связана с величиной априорной вероятности получения этого сигнала. В частности, полезная информация не уменьшается при увеличении вероятности получить сигнал «вперед».

Интуитивно ясно, что несоответствие полезной информации, содержащейся в сигнале, и разнообразия различимых сигналов объясняется тем, что система еще до прихода сигнала обладала запасом полезной информации в декодирующем алгоритме (пакет!). Пришедший сигнал перевел эту информацию из «скрытого» состояния в распределение $\boldsymbol{q}$ решающего алгоритма. В таком виде информация уже проявляется при решении задачи.

Дадим теперь формальное определение запаса полезной информации в декодирующем устройстве (мы будем его также называть *скрытой* информацией). Обратимся к таблице 2, характеризующей декодирующий алгоритм. Пусть

$$D_i = \frac{1}{m} \sum_{k=1}^{m} d_{i,k}. \tag{19}$$

Тогда скрытой информацией мы будем называть:

$$I_{\text{скр}} = \log n - N(\boldsymbol{p}/\boldsymbol{D}). \tag{20}$$

В случае $I_{\text{скр}} = 0$ мы будем говорить о системе без скрытой информации (относительно данной задачи) *).

Рассмотрим соотношение между скрытой информацией, пропускной способностью канала связи и запасом полезной информации, который может быть создан в решающем алгоритме в результате декодирования сигнала.

*) Достаточным условием отсутствия в системе скрытой информации является равенство между собой всех сумм по строкам таблицы 2. При этом $D_i = \frac{1}{n}$ и $I_{\text{скр}}$ относительно любой задачи.

Пусть в результате посылки некоторого сообщения θ на приемном конце канала связи с вероятностью P_1 появляется сигнал C_1, с вероятностью P_2 — сигнал C_2 и т. д.*). Неопределенность, которую после прихода такого сообщения будет в среднем иметь задача $\boldsymbol{A}$ с распределением вероятностей ответов $\boldsymbol{p}$, определяется следующим образом:

$$N(\boldsymbol{A}) = -\sum_{k=1}^{m} P_k \sum_{i=1}^{n} p_i \log d_{i,k}. \tag{21}$$

Преобразуем (21):

$$N(\boldsymbol{A}) = -\sum_{i=1}^{n} p_i \sum_{k=1}^{m} P_k \log d_{i,k} = -\sum_{i=1}^{n} p_i \sum_{k=1}^{m} P_k \left(\log D_i + \log m + \log \frac{d_{i,k}}{mD_i} \right).$$

Так как $\sum_{i=1}^{n} p_i = 1$ и $\sum_{k=1}^{m} P_k = 1$, то

$$N(\boldsymbol{A}) = -\log m - \sum_{i=1}^{n} p_i \left(\log D_i + \sum_{k=1}^{m} P_k \log \frac{d_{i,k}}{mD_i} \right). \tag{22}$$

Из соотношения (19) следует $\sum_{k=1}^{m} \frac{d_{i,k}}{mD_i} = 1$. Учитывая это и соотношение (4), оценим снизу выражение $-\sum_{k=1}^{m} P_k \log \frac{d_{i,k}}{mD_i}$

$$-\sum_{k=1}^{m} P_k \log \frac{d_{i,k}}{mD_i} \geqslant -\sum_{k=1}^{m} P_k \log P_k = H(\boldsymbol{P}). \tag{23}$$

Через $H(\boldsymbol{P})$ мы обозначили энтропию распределения вероятностей «разброса» приходящих сигналов C_k при условии посылки сообщения θ (но не вероятностей посылки разных сообщений!).

Теперь мы можем оценить снизу неопределенность. Учитывая (22) и (23), получим:

$$N(\boldsymbol{A}) \geqslant -\log m - \sum_{i=1}^{n} p_i [\log D_i - H(\boldsymbol{P})] = H(\boldsymbol{P}) - \log m + N(\boldsymbol{p}/\boldsymbol{D}), \tag{24}$$

или, принимая во внимание (20),

$$N(\boldsymbol{A}) \geqslant H(\boldsymbol{P}) - \log m + \log n - I_{\text{скр}}. \tag{25}$$

Соотношение (25) можно записать так:

$$\log n - N(\boldsymbol{A}) \leqslant [\log m - H(\boldsymbol{P})] + I_{\text{скр}}. \tag{26}$$

В левой части неравенства (26) стоит запас полезной информации в *решающем* алгоритме**) после прихода сообщения. В квадратных скобках — пропускная способность канала на одно сообщение. Таким образом, если учитывать скрытую информацию, то увеличение полезной информации не будет парадоксальным.

*) Нас сейчас не интересует причина «разброса» приходящих сигналов. Это может быть влияние помех, закона работы кодирующего сигнала алгоритма и т. п.

**) Можно также говорить, что это — полезная информация, содержащаяся в сигнале для наблюдателя, «ничего не знающего» о задаче.

Формула (26) накладывает ограничения на запас полезной информации в *решающем* алгоритме, а не на ее прирост *) после прихода сообщения. Поэтому если канал имеет небольшую пропускную способность на одно сообщение, а мы хотим накопить в системе большой запас полезной информации путем посылки серии сообщений, то сделать это можно лишь путем запасания информации в *декодирующем* алгоритме. Этот процесс заключается в том, что после декодирования сообщения новому распределению q' соответствует новый декодирующий алгоритм с новым запасом скрытой информации. На следующем шаге скрытая информация снова повлияет на q'', что опять изменит скрытую информацию, и т. д.

Возникает вопрос: почему мы определили полезную информацию, заключенную в *сигнале*, как изменение неопределенности задачи? Не лучше ли было отделить полезную информацию, перешедшую в решающий алгоритм из декодирующего устройства, от полезной информации, пришедшей по каналу связи, и лишь последнюю считать полезной информацией, заключенной в сигнале?

Для ответа на этот вопрос обратим внимание на некоторую непоследовательность, проявленную при определении понятия «скрытая информация». До этого момента мы все время говорили о свойствах *индивидуального* сигнала. Изменение неопределенности задачи зависит от данного сигнала (и состояния системы) и вовсе не зависит от того, какие еще сигналы *могли бы* прийти. При определении же скрытой информации мы явно использовали некоторое *среднее свойство совокупности возможных сигналов*. Формулу (20) можно интерпретировать так: перережем линию связи и с помощью жребия будем на приемном конце генерировать сигналы C_k. После каждого сигнала замеряем q' и возвращаем систему в прежнее состояние. После многих опытов находим средние q'_i. Для решения задачи пользуемся этими средними q'_i. Если такой «средний сигнал» сделал неопределенность отличной от $\log n$, мы говорим о наличии в системе скрытой информации. Теперь видно, что формулы (19) и (20) в совокупности являются определением скрытой информации в некотором частном случае — если считать равновероятными приходы всех различимых сигналов. При другой гипотезе о вероятностях прихода C_k мы должны приписать *той же системе* другой запас скрытой информации.

Оценим конечную неопределенность задачи в случае произвольной гипотезы о вероятностях прихода сигналов C_k. Формула (20) в этом случае сохраняет свою силу, а формула (19) переходит в формулу

$$D_i = \sum_{k=1}^{m} \overline{Q}_k d_{i,k}, \tag{27}$$

где $\overline{Q}_k$ — ожидаемая вероятность прихода C_k.

$$N(\boldsymbol{A}) = -\sum_{k=1}^{m} P_k \sum_{i=1}^{n} p_i \log d_{i,k} = -\sum_{i=1}^{n} p_i \sum_{k=1}^{m} P_k \Big(\log D_i - \log \overline{Q}_k + \log \frac{d_{i,k}\overline{Q}_k}{D_i} \Big) = N(\boldsymbol{p}/\boldsymbol{D}) - N(\boldsymbol{P}/\overline{\boldsymbol{Q}}) - \sum_{i=}^{n} p_i \sum_{k=1}^{m} P_k \log \frac{d_{i,k}\overline{Q}_k}{D_i}. \tag{28}$$

Принимая во внимание (28), (27), (20) и (4), можем написать:

$$N(\boldsymbol{A}) \geqslant H(\boldsymbol{P}) - N(\boldsymbol{P}/\overline{\boldsymbol{Q}}) + \log n - I_{\text{скр}},$$

*) Уменьшение неопределенности может быть очень большим, если начальная неопределенность была больше $\log n$. В этом нет ничего неожиданного, так как уменьшения неопределенности до $\log n$ всегда можно достигнуть путем *увеличения* энтропии распределения q_i, а следовательно, без прихода информации извне.

или

$$\log n - N(A) \leqslant [N(P/\overline{Q}) - H(P)] + I_{\text{скр}}. \tag{29}$$

Нам кажется целесообразным считать выражение, стоящее в квадратных скобках, *средней пропускной способностью канала* при распределении вероятностей прихода сигналов P и гипотезе о распределении этих вероятностей $\overline{Q}$.

При изменении распределения $\overline{Q}$ изменяется пропускная способность канала, но одновременно изменяется и $I_{\text{скр}}$. Поэтому деление полезной информации, приобретенной решающим алгоритмом, на информацию, пришедшую по каналу связи, и информацию, извлеченную из декодирующего устройства, зависит от исходной гипотезы о вероятностях сигналов.

Изменение же неопределенности задачи от этой гипотезы не зависит и полностью определяется данным *единичным* сообщением. Поэтому мы и определили полезную информацию, содержащуюся в данном сигнале, через изменение неопределенности задачи.

Обратим внимание на выражение, стоящее в квадратных скобках в формуле (29). Оно характеризует степень «неожиданности» приходящих сигналов. Если $\overline{Q}_k = P_k$, то в среднем мы из приходящего сигнала ничего извлечь не можем (мы могли бы сами его генерировать на приемном конце канала с помощью подходящего по распределению вероятностей случайного процесса), а используем лишь скрытую информацию.

§ 5. Построение декодирующих алгоритмов

Формула (29) ограничивает сверху возможности декодирующих алгоритмов. В частности, любое устройство, не обладающее относительно ожидаемого распределения вероятностей сигналов $\overline{Q}_k$ запасом скрытой информации, в среднем даст результат

$$\log n - N(A) \leqslant N(P/\overline{Q}) - H(P). \tag{30}$$

Покажем теперь, что если

$$\frac{n\overline{Q}_k}{P_k} \geqslant 1 \quad \text{для всех} \quad k, \tag{31}$$

то для задачи с распределением вероятностей ответов p всегда можно построить декодирующий алгоритм, не имеющий скрытой информации и дающий в среднем после посылки сообщения θ с распределением вероятностей сигналов P запас полезной информации в решающем алгоритме

$$\log n - N(A) \geqslant N(P/\overline{Q}) - H(P) - H(p).$$

Построим таблицу $d_{i,k}$ следующим образом:

$$d_{i,k} = \frac{p_i P_k}{n\overline{Q}_k} + \frac{(1-p_i)(n\overline{Q}_k - P_k)}{n(n-1)\overline{Q}_k}. \tag{32}$$

Покажем, что эта таблица определяет некоторый декодирующий алгоритм, не имеющий скрытой информации относительно распределения вероятностей сигналов $\overline{Q}_k$. Действительно, $d_{i,k} \geqslant 0$, так как в силу (31)

$$\frac{(1-p_i)(n\overline{Q}_k - P_k)}{n(n-1)\overline{Q}_k} \geqslant 0. \tag{33}$$

Кроме того, несложными преобразованиями можно убедиться, что $\sum_{i=1}^{n} d_{i,k} = 1$, а $\sum_{k=1}^{m} \bar{Q}_k d_{i,k} = \frac{1}{n}$. Последнее соотношение является достаточным условием отсутствия в системе скрытой информации (см. сноску на стр. 83). Вопрос о том, будут ли все столбцы различными, мы не ставим, так как не стремимся к тому, чтобы система различала все сигналы C_k.

Оценим среднюю неопределенность задачи после декодирования сообщения построенным алгоритмом:

$$N(\boldsymbol{A}) = -\sum_{k=1}^{m} P_k \sum_{i=1}^{n} p_i \log d_{i,k} =$$

$$= -\sum_{k=1}^{m} \sum_{i=1}^{n} P_k p_i \log \left[\frac{p_i P_k}{n\bar{Q}_k} + \frac{(1-p_i)(n\bar{Q}_k - P_k)}{n(n-1)\bar{Q}_k} \right].$$

Учитывая соотношение (33), можно написать:

$$N(\boldsymbol{A}) \leqslant -\sum_{k=1}^{m} \sum_{i=1}^{n} P_k p_i \log \frac{p_i P_k}{n\bar{Q}_k} = \log n + H(\boldsymbol{P}) + H(\boldsymbol{p}) - N(\boldsymbol{P}/\bar{\boldsymbol{Q}}),$$

откуда окончательно получаем:

$$\log n - N(\boldsymbol{A}) \geqslant N(\boldsymbol{P}/\bar{\boldsymbol{Q}}) - H(\boldsymbol{P}) - H(\boldsymbol{p}). \tag{34}$$

В случае $H(\boldsymbol{p}) = 0$ формула (34) переходит в

$$\log n - N(\boldsymbol{A}) \geqslant N(\boldsymbol{P}/\bar{\boldsymbol{Q}}) - H(\boldsymbol{P}). \tag{35}$$

Сравнивая формулы (35) и (30), мы видим, что в случае $H(\boldsymbol{p}) = 0$ указанный способ построения декодирующего алгоритма дает наилучший возможный (для алгоритма без скрытой информации) результат:

$$\log n - N(\boldsymbol{A}) = N(\boldsymbol{P}/\bar{\boldsymbol{Q}}) - H(\boldsymbol{P}).$$

В общем случае мы не знаем, является ли этот способ оптимальным. Гарантируется лишь, что если $H(\boldsymbol{p}) > 0$, то в формуле (34) имеет место строгое неравенство.

Легко показать, что построить декодирующий алгоритм указанным способом удается только при соблюдении условия (31). Смысл его в том, что сигналы, приходящие в действительности часто, нельзя считать очень мало вероятными.

* * *

Пусть теперь имеется r задач $\boldsymbol{A}_1, \boldsymbol{A}_2, \ldots, \boldsymbol{A}_l, \ldots, \boldsymbol{A}_r$ и вероятности ответов задач задаются матрицей $\| p_{i,l} \|$ (см. таблицу 3).

Таблица 3

	$\boldsymbol{A}_1$	$\boldsymbol{A}_2$	...	$\boldsymbol{A}_l$	...	$\boldsymbol{A}_r$
b_1	$p_{1,1}$	$p_{1,2}$	...	$p_{1,l}$	...	$p_{1,r}$
...	...	...	...	...	...	...
b_i	$p_{i,1}$	$p_{i,2}$	...	$p_{i,l}$	...	$p_{i,r}$
...	...	...	...	...	...	...
b_n	$p_{n,1}$	$p_{n,2}$	...	$p_{n,l}$	...	$p_{n,r}$

О необходимости решать задачу A_l нас извещают путем посылки сообщения θ_l. Вероятности прихода сигналов C_k после посылки различных сообщений определяются матрицей $\| P_{l,k} \|$ (см. таблицу 4).

Таблица 4

	C_1	C_2	...	C_k	...	C_m
θ_1	$P_{1,1}$	$P_{1,2}$	...	$P_{1,k}$	...	$P_{1,m}$
...	...	...	...	...	...	...
θ_l	$P_{l,1}$	$P_{l,2}$	...	$P_{l,k}$	...	$P_{l,m}$
...	...	...	...	...	...	...
θ_r	$P_{r,1}$	$P_{r,2}$	...	$P_{r,k}$	...	$P_{r,m}$

Очевидно, $\sum_{i=1}^{n} p_{i,l} = 1$ и $\sum_{k=1}^{m} P_{l,k} = 1$.

Пусть все сигналы декодируются одним и тем же алгоритмом с матрицей $\| d_{i,k} \|$ (см. таблицу 2).

Обозначим через $\overline{N}(A)$ среднюю неопределенность задачи после посылки сообщения. Если задача A_l встречается с вероятностью R_l, причем $\sum_{l=1}^{r} R_l = 1$, то положим:

$$\overline{N}(A) = -\sum_{l=1}^{r} R_l \sum_{k=1}^{m} P_{l,k} \sum_{i=1}^{n} p_{i,l} \log d_{i,k}. \tag{36}$$

Введем обозначения:

$$\overline{P}_k = \sum_{l=1}^{r} R_l P_{l,k}, \tag{37}$$

$$\overline{p}_i = \sum_{l=1}^{r} R_l p_{i,l}, \tag{38}$$

$$\overline{d}_i = \sum_{k=1}^{m} \overline{P}_k d_{i,k}. \tag{39}$$

Легко показать, что

$$\sum_{k=1}^{m} \overline{P}_k = 1, \tag{40}$$

$$\sum_{i=1}^{n} \overline{p}_i = 1, \tag{41}$$

$$\sum_{i=1}^{n} \overline{d}_i = 1, \tag{42}$$

$$\sum_{i=1}^{n} \sum_{l=1}^{r} \frac{R_l P_{l,k} p_{i,l}}{\overline{P}_k} = 1. \tag{43}$$

Для оценки снизу средней неопределенности преобразуем (36):

$$\overline{N}(\boldsymbol{A}) = -\sum_{l=1}^{r} R_l \sum_{k=1}^{m} P_{l,k} \sum_{i=}^{n} p_{i,l}\left(\log \overline{d}_i - \log \overline{P}_k + \log \frac{\overline{P}_k d_{i,k}}{\overline{d}_i}\right) =$$

$$= \sum_{l=1}^{r} R_l \sum_{k=1}^{m} P_{l,k} \log \overline{P}_k - \sum_{l=1}^{r} R_l \sum_{i=1}^{n} p_{i,l} \log \overline{d}_i -$$

$$- \sum_{l=1}^{r} R_l \sum_{i=1}^{n} p_{i,l} \sum_{k=1}^{m} P_{l,k} \log \frac{\overline{P}_k d_{i,k}}{\overline{d}_i} = -H(\overline{\boldsymbol{P}}) + N(\overline{\boldsymbol{p}}/\overline{\boldsymbol{d}}) -$$

$$- \sum_{l=1}^{r} R_l \sum_{i=1}^{n} p_{i,l} \sum_{k=1}^{m} P_{l,k} \log \frac{\overline{P}_k d_{i,k}}{\overline{d}_i}.$$

Учитывая (39) и (4), получаем:

$$\overline{N}(\boldsymbol{A}) \geqslant N(\overline{\boldsymbol{p}}/\overline{\boldsymbol{d}}) - H(\overline{\boldsymbol{P}}) + \overline{H}(\boldsymbol{P}), \tag{44}$$

где $\overline{H}(\boldsymbol{P}) = -\sum_{l=1}^{r} R_l \sum_{k=1}^{m} P_{l,k} \log P_{l,k}$ есть средняя энтропия распределения вероятностей сигналов C_k в сообщениях θ_l.

Из (44) следует также

$$\overline{N}(\boldsymbol{A}) \geqslant H(\overline{\boldsymbol{p}}) - H(\overline{\boldsymbol{P}}) + \overline{H}(\boldsymbol{P}). \tag{45}$$

Посмотрим, как должна быть построена матрица декодирующего алгоритма, если мы хотим получить минимум средней неопределенности задачи после декодирования сигналов. Запишем (36) в виде

$$\overline{N}(\boldsymbol{A}) = -\sum_{k=1}^{m} \overline{P}_k \left[\sum_{i=1}^{n} \left(\sum_{l=1}^{r} \frac{R_l P_{l,k} p_{i,l}}{\overline{P}_k}\right) \log d_{i,k}\right].$$

Принимая во внимание (43) и то, что $\sum_{i=1}^{n} d_{i,k} = 1$, мы видим, что выражение в квадратных скобках, а с ним и средняя неопределенность минимальны при

$$d_{i,k} = \sum_{l=1}^{r} \frac{R_l P_{l,k} p_{i,l}}{\overline{P}_k}. \tag{46}$$

Легко убедиться, что при построении матрицы $d_{i,k}$ по формуле (46) имеет место равенство $\overline{d}_i = \overline{p}_i$. Это значит, что оптимальный декодирующий алгоритм обладает максимально возможной скрытой информацией относительно «средней задачи» (среднего распределения вероятностей ответов для всех задач).

Оценим среднюю неопределенность задачи после применения оптимального декодирующего алгоритма. Так как

$$d_{i,k} \geqslant \frac{R_l P_{l,k} p_{i,l}}{\overline{P}_k},$$

то

$$\overline{N}(\boldsymbol{A}) \leqslant -\sum_{l=1}^{r} R_l \sum_{k=1}^{m} P_{l,k} \sum_{i=1}^{n} p_{i,l} \log \frac{R_l P_{l,k} p_{i,l}}{\overline{P}_k} =$$

$$= H(\boldsymbol{R}) + \overline{H}(\boldsymbol{P}) - H(\overline{\boldsymbol{P}}) - \sum_{l=1}^{r} R_l \sum_{i=1}^{n} p_{i,l} \log p_{i,l},$$

т. е.

$$\overline{N}(\boldsymbol{A}) \leqslant H(\boldsymbol{R}) + \overline{H}(\boldsymbol{P}) - H(\overline{\boldsymbol{P}}) + \overline{H}(\boldsymbol{p}), \tag{47}$$

где

$$\overline{H}(\boldsymbol{p}) = -\sum_{l=1}^{r} R_l \sum_{i=1}^{n} p_{i,l} \log p_{i,l}.$$

Формула (47) оценивает неопределенность после декодирования сигнала оптимальным способом.

Представим себе теперь, что декодирующий алгоритм конструирует наблюдатель, исходящий из гипотез:

а) вероятность появления l-й задачи равна S_l;

б) вероятности ответов разных задач соответствуют матрице $q_{i,l}$;

в) вероятности прихода сигналов C_k после посылки различных сообщений определяются матрицей $\| Q_{l,k} \|$.

Введем обозначения, аналогичные (37), (38) и (39):

$$\overline{Q}_k = \sum_{l=1}^{r} S_l Q_{l,k}, \tag{48}$$

$$\overline{q}_i = \sum_{l=1}^{r} S_l q_{i,l}, \tag{49}$$

$$\overline{d}'_i = \sum_{k=1}^{m} \overline{Q}_k d_{i,k}. \tag{50}$$

Стремясь получить наилучший результат, наш наблюдатель, естественно, выберет

$$d'_{i,k} = \sum_{l=1}^{r} \frac{S_l Q_{l,k} q_{i,l}}{\overline{Q}_k}. \tag{51}$$

Средняя неопределенность задачи после такого декодирования будет:

$$\overline{N}(\boldsymbol{A}) = \sum_{l=1}^{r} R_l \sum_{k=1}^{m} P_{l,k} \sum_{i=1}^{n} p_{i,l} \log d'_{i,k}. \tag{52}$$

С помощью подстановки, аналогичной той, которая применялась при выводе формулы (44), легко получить:

$$\overline{N}(\boldsymbol{A}) \geqslant N(\overline{\boldsymbol{p}}/\overline{\boldsymbol{d}}') - N(\overline{\boldsymbol{P}}/\overline{\boldsymbol{Q}}) + \overline{H}(\boldsymbol{P}). \tag{53}$$

Из (51) следует $\overline{d}'_i = \overline{q}_i$, поэтому

$$N(\overline{\boldsymbol{p}}/\overline{\boldsymbol{q}}) - \overline{N}(\boldsymbol{A}) \leqslant N(\overline{\boldsymbol{P}}/\overline{\boldsymbol{Q}}) - \overline{H}(\boldsymbol{P}). \tag{54}$$

Посмотрим теперь, чему равна средняя *начальная* неопределенность $\overline{N}_0(\boldsymbol{A})$ задач для нашего наблюдателя. Допустим, что до прихода

сигналов он пользуется решающим алгоритмом с распределением вероятностей $\boldsymbol{x}$; тогда

$$\overline{N}_0(\boldsymbol{A}) = -\sum_{l=1}^{r} R_l \sum_{i=1}^{n} p_{i,\,l} \log x_i = N(\overline{\boldsymbol{p}}/\boldsymbol{x}). \tag{55}$$

Начальная неопределенность достигает минимума при $x_i = \overline{p}_i$, поэтому наблюдатель положит $x_i = \overline{q}_i$. Итак, средняя начальная неопределенность при данной гипотезе определяется формулой

$$\overline{N}_0(\boldsymbol{A}) = N(\overline{\boldsymbol{p}}/\overline{\boldsymbol{q}}). \tag{56}$$

Сопоставляя с (54), получаем:

$$\overline{N}_0(\boldsymbol{A}) - \overline{N}(\boldsymbol{A}) \leqslant N(\overline{\boldsymbol{P}}/\overline{\boldsymbol{Q}}) - \overline{H}(\boldsymbol{P}). \tag{57}$$

В левой части неравенства (57) стоит средняя полезная информация, пришедшая по каналу связи, поэтому правую часть естественно считать пропускной способностью канала связи для *полезной* информации при решении нескольких задач. Для наблюдателя, *все знающего правильно* о задачах и о канале связи ($S_l = R_l$; $q_{i,\,l} = p_{i,\,l}$; $Q_{l,\,k} = P_{l,\,k}$), получается обычная граница $H(\overline{\boldsymbol{P}}) - \overline{H}(\boldsymbol{P})$.

Декодируя сигнал оптимальным со своей точки зрения способом, наблюдатель всегда может получить результат

$$\overline{N}(\boldsymbol{A}) \leqslant N(\boldsymbol{R}/\boldsymbol{S}) + \overline{N}(\boldsymbol{p}/\boldsymbol{q}) + \overline{N}(\boldsymbol{P}/\boldsymbol{Q}) - N(\overline{\boldsymbol{P}}/\overline{\boldsymbol{Q}}), \tag{58}$$

где черта над функцией всюду указывает на усреднение по разным задачам (по l) с распределением вероятностей $\boldsymbol{R}$ (реальным, а не гипотетическим). Формула (58) выводится аналогично формуле (47).

При выводе формул (56) и (57) мы считали, что начальное состояние решающего алгоритма (распределение q_i до прихода сообщения) не произвольно, а определяется гипотезой наблюдателя об относительной частоте разных задач $\boldsymbol{A}_l$ и о распределении вероятностей их ответов (S_l и $q_{i,\,l}$).

Если бы мы так же поступили при выводе формулы (29), то оказалось бы, что «последовательный наблюдатель», исходящий из гипотезы о распределении вероятностей сигналов $\overline{\boldsymbol{Q}}$ и разумности своего декодирующего устройства, должен перевести решающий алгоритм до прихода сообщения в состояние $q_i = D_i$. Мы получили бы формулу

$$N(\boldsymbol{p}/\boldsymbol{D}) - N(\boldsymbol{A}) \leqslant N(\boldsymbol{P}/\overline{\boldsymbol{Q}}) - H(\boldsymbol{P}). \tag{57'}$$

В левой части неравенства (57′) стоит прирост полезной информации (для «последовательного наблюдателя»). При таком подходе нет необходимости вводить понятие «скрытая информация», и прирост полезной информации не превышает пропускной способности канала. Однако это достигается ценой уменьшения степеней свободы гипотез, которыми может пользоваться наблюдатель. По-видимому, оба подхода являются равноправными.

§ 6. Частные задачи с несколькими решениями

В этом параграфе мы снимаем одно из ограничений, введенных в § 2, и рассматриваем частные задачи, имеющие *не менее одного* решения. Это значит, что $\sum_{i=1}^{n} p_i \geqslant 1$.

Неопределенностью задачи $\boldsymbol{A}$ мы по-прежнему будем называть математическое ожидание $\log \overline{K}$, где $\overline{K}$—среднее число проб, необходимых

для нахождения *какого-нибудь одного* (а не всех!) решения частной задачи a_j.

Теперь формула (3) уже не дает возможности по вероятностям p_i и q_i найти неопределенность задачи. Проиллюстрируем это примером. Пусть все a_j имеют по s ответов. При равномерном распределении вероятностей $p_i = \frac{s}{n}$, $i = 1, 2, \ldots, n$. Пусть решающий алгоритм пользуется распределением вероятностей $q_i = \frac{1}{n}$, $i = 1, 2, \ldots, n$. Как легко сообразить, среднее число проб для этого алгоритма будет n/s, откуда $N(\boldsymbol{A}) = \log(n/s)$. Если же подставить p_i и q_i в формулу (3), то получится $s \log n$.

В общем случае формула для неопределенности имеет вид

$$-N(\boldsymbol{A}) = \sum_i r_i \log q_i + \sum_{i,k} r_{i,k} \log(q_i + q_k) + \sum_{i,k,l} r_{i,k,l} \log(q_i + q_k + q_l) + \ldots$$

$$\ldots + \sum_{i,k,\ldots,m} r_{i,k,\ldots,m} \log(\overbrace{q_i + q_k + \ldots + q_m}^{n-1}), \quad (59)$$

где r_i — вероятность того, что выбирается частная задача, которая имеет решение b_i, причем оно является *единственным* решением задачи, $r_{i,k}$ — вероятность того, что b_i и b_k одновременно являются решениями и других решений нет, и т. д. Смысл q_i остается прежним, в частности $\sum_{i=1}^{n} q_i = 1$. Суммы в (59) берутся при $i \neq k$; $i \neq l$; $\ldots$; $i \neq m$; $k \neq l$; $\ldots$

Для $n = 2$

$$N(\boldsymbol{A}) = -(1 - p_2)\log q_1 - (1 - p_1)\log q_2.$$

Неопределенность в этом случае достигает минимума при $q_1 = \frac{1 - p_2}{2 - p_1 - p_2}$; $q_2 = \frac{1 - p_1}{2 - p_1 - p_2}$. Если энтропию задачи считать *по определению равной минимуму неопределенности*, то при $n = 2$ и $p_1 + p_2 \geqslant 1$

$$H(\boldsymbol{p}) = (2 - p_1 - p_2)\log(2 - p_1 - p_2) - (1 - p_1)\log(1 - p_1) - (1 - p_2)\log(1 - p_2). \quad (60)$$

Таким образом, можно получить некоторое обобщение понятия энтропии, не имевшей определения для случая $\Sigma p_i > 1$.

§ 7. Системы с обратной связью

Теперь мы снимем второе ограничение, введенное в § 2 и заключающееся в стабильности q_i в процессе решения задачи. Решающие алгоритмы, которые рассматривались до сих пор, «ничего не помнили». После того как применение W-алгоритма показало, что, например, b_1 не является ответом, решающий алгоритм продолжал бросать жребий со старым распределением вероятностей $\boldsymbol{q}$, хотя уже стало известно, что q_1 можно сделать равным нулю.

Этот недостаток можно устранить, например, следующим образом: соединим W-алгоритм со входом декодирующего устройства. Пусть в случае неудачной пробы по цепи обратной связи посылается сигнал «не b_k». Получив такое сообщение, декодирующее устройство делает $q'_k = 0$, а остальные вероятности нормирует к единице. Теперь после каждой неудачной пробы неопределенность задачи уменьшается. Система с обратной связью пользуется на каждом следующем шаге новой (уточненной в результате эксперимента) гипотезой. Поэтому в общем случае формулы,

выражающие неопределенность, для систем с обратной связью получаются очень громоздкими.

В частном случае, если решающий алгоритм использует на первом шаге $q_i^1 = \frac{1}{n}$, на втором шаге $q_i^2 = \frac{1}{n-1}$ для всех i, кроме одного, отвергнутого первым опытом и т. д., неопределенность при $\sum_{i=1}^{n} p_i = 1$ имеет вид *)

$$N(\boldsymbol{A}) = \log \frac{n+1}{2}.$$

Если частные задачи имеют s ответов, то неопределенность задачи для рассматриваемого алгоритма равна $\log \frac{n+1}{s+1}$, а сокращение неопределенности по сравнению с системой без обратной связи равно

$$\Delta N(\boldsymbol{A}) = \log \frac{n}{s} - \log \frac{n+1}{s+1} = \log \left(\frac{n}{n+1} \cdot \frac{s+1}{s} \right) < \log 2.$$

При увеличении s (разумеется, всегда $s \leqslant n$) выигрыш стремится к нулю. Причина этого понятна; при большом s ответ в среднем бывает уже найден, когда «перепробована» лишь малая часть всех b_i. До самого конца процесса решения неопределенность задачи не успевает заметно измениться, и средняя неопределенность мало отличается от начальной.

Рассмотренная примитивная обратная связь, сводящаяся к выкидыванию проверенных b_k, уменьшает неопределенность задачи для алгоритмов с равномерным распределением вероятностей гипотезы не более чем на $\log 2$. Однако для гипотез с произвольными $\boldsymbol{q}$ экономия может быть сколь угодно большой. Применение системы с обратной связью особенно выгодно тогда, когда начальная гипотеза очень сильно не соответствует действительному распределению $\boldsymbol{p}$.

В § 2 было показано, что при построении решающего алгоритма *без обратной связи* (с постоянным от шага к шагу распределением $\boldsymbol{q}$) наиболее выгодным является бросать жребий с распределением вероятностей $\boldsymbol{p}$. Однако это вовсе не означает, что тот же способ будет наилучшим и для решающего алгоритма с обратной связью. И действительно, как будет показано в § 8, для случая $\Sigma p_i = 1$ оптимальным будет следующий решающий алгоритм с обратной связью. Упорядочим индексы при b так, чтобы $p_{l_1} \geqslant p_{l_2} \geqslant \ldots \geqslant p_{l_n}$. Пробуем b_{l_1}; если ответ «ложно», то требуем b_{l_2} и т. д. Таким образом, оптимальный решающий алгоритм с обратной связью фактически обходится без жребия (все выборы происходят с вероятностями единица). Каждый ответ b_i характеризуется для этого алгоритма не вероятностью выбора q_i, а натуральным числом z_i — порядковым номером в «очереди на проверку».

При повторных решениях частной задачи a_j, имеющей ответ b_i, решающий алгоритм с жестким порядком перебора будет всегда находить ответ на одном и том же шаге z_i. Поэтому неопределенность $N(a_j)$ для такого решающего алгоритма равна просто $\log z_i$. Отсюда

$$N(\boldsymbol{A}) = \sum_{i=1}^{n} p_i \log z_i = \sum_{s=1}^{n} p_{l_s} \log z_{l_s} = \sum_{s=1}^{n} p_{l_s} \log s, \tag{61}$$

где s — номер p_{l_s} в ряду p_i, расставленных в порядке убывания.

*) Среднее число проб при работе данного алгоритма

$$\bar{K} = \frac{1}{n}(1+2+\ldots+n) = \frac{n+1}{2}.$$

Естественно, что если при построении решающего алгоритма мы исходим из гипотезы $\boldsymbol{q}$ о распределении вероятностей ответов, то в порядке убывания будут расположены q_i. Соответственно в формуле (61) s нужно считать номером q_{t_s} в убывающем ряду q_i.

Докажем, что в этом случае

$$N(\boldsymbol{A}) = \sum_{s=1}^{n} p_{t_s} \log s \leqslant N(\boldsymbol{p}/\boldsymbol{q}). \tag{62}$$

Заметим, что имеют место соотношения

$$\begin{aligned} 1 &\geqslant q_{t_1}, \\ \frac{1}{2} &\geqslant q_{t_2}, \\ \frac{1}{3} &\geqslant q_{t_3}, \\ &\cdots\cdots \\ \frac{1}{n} &\geqslant q_{t_n}. \end{aligned}$$

Следовательно,

$$\begin{aligned} \log 1 &\leqslant -\log q_{t_1}, \\ \log 2 &\leqslant -\log q_{t_2}, \\ &\cdots\cdots\cdots \\ \log n &\leqslant -\log q_{t_n}, \end{aligned}$$

откуда формула (62) получается автоматически.

Допустим, мы имеем два решающих алгоритма с жестким порядком перебора. Один исходит из гипотезы $\boldsymbol{q}$, а другой — из гипотезы $\boldsymbol{q}'$. Допустим, что $q_i \neq q_i'$, но $z_i = z_i'$. Мы не сможем по «поведению» отличить один алгоритм от другого. Поэтому имеет смысл говорить, что для алгоритмов с обратной связью гипотезой является не распределение вероятностей, а лишь порядок проверки ответов.

Декодирующее устройство, связанное с таким решающим алгоритмом, целесообразно характеризовать таблицей, определяющей очередь проверки ответа b_i после прихода сигнала C_k (см. таблицу 5).

Таблица 5

	C_1	C_2	$\ldots$	C_k	$\ldots$	C_m
b_1	$z_{1,1}$	$z_{1,2}$	$\cdots$	$z_{1,k}$	$\cdots$	$z_{1,m}$
b_2	$z_{2,1}$	$z_{2,2}$	$\cdots$	$z_{2,k}$	$\cdots$	$z_{2,m}$
$\ldots$	$\ldots$	$\ldots$	$\ldots$	$\ldots$	$\ldots$	$\ldots$
b_i	$z_{i,1}$	$z_{i,2}$	$\cdots$	$z_{i,k}$	$\cdots$	$z_{i,m}$
$\ldots$	$\ldots$	$\ldots$	$\ldots$	$\ldots$	$\ldots$	$\ldots$
b_n	$z_{n,1}$	$z_{n,2}$	$\cdots$	$z_{n,k}$	$\cdots$	$z_{n,m}$

В столбце $z_{1,k}, z_{2,k}, \ldots, z_{n,k}$ все целые числа от 1 до n встречаются по одному разу.

Очевидно, оптимальным будет декодирующее устройство $z_{i,k}$, построенное следующим образом. Возьмем матрицу $\| d_{i,k} \|$, где $d_{i,k}$ определяется по формуле (46). Расставим элементы k-го столбца в порядке убывания: $d_{t_1,k} \geqslant d_{t_2,k} \geqslant \ldots \geqslant d_{t_n,k}$. Пусть $i = t_s$, тогда $z_{i,k} = s$.

Учитывая соотношение (62), легко заметить, что формула (47) справедлива и для средней неопределенности задач по отношению к построенной нами системе с жестким порядком перебора после декодирования.

* *
*

Существуют системы, в которых можно исследовать приходящий извне сигнал, можно сосчитать число проб (применений W-алгоритма), приводящих к отысканию ответа некоторой задачи, но пока что невозможно проследить процесс декодирования сигнала. Примером такой системы может служить животный организм.

Существуют также системы, в которых хотя и возможно описать алгоритм декодирования сигнала, но очень трудно выяснить, насколько он близок к идеальному. К таким системам относятся некоторые программы для цифровых машин.

В случае затруднений, возникающих у таких систем при решении задач, можно ставить вопрос об их причине. Оттого ли нужно много попыток, что мала поступающая извне информация, или дело в плохой работе декодирующего устройства?

При решении этого вопроса иногда может помочь формула (47). Действительно, средняя неопределенность задач может быть определена экспериментально:

$$\overline{N}(\boldsymbol{A}) = \overline{\log K}, \tag{63}$$

где K — число проб, необходимых системе для решения частной задачи после декодирования соответствующего сигнала, а среднее берется для разных задач и соответствующих сообщений. Естественно, эксперимент должен быть достаточно обширным.

В случае нарушения неравенства

$$\overline{\log K} \leqslant H(\boldsymbol{R}) + \overline{H}(\boldsymbol{P}) - H(\overline{\boldsymbol{P}}) + \overline{H}(\boldsymbol{p}) \tag{64}$$

можно утверждать, что декодирующее устройство является неидеальным.

В этом случае заведомо можно построить другое декодирующее устройство, которое для тех же задач ($p_{i,l}$), той же системы кодирования ($P_{l,k}$) и тех же вероятностей задач (R_l) даст меньшую неопределенность. Такой вывод может оказаться полезным при поисках пути усовершенствования некоторых программ. В частности, поиск может вести и сама программа.

При исследовании поведения животного в случае нарушения (64) можно предположить, что его «декодирующее устройство» приспособлено не к «миру $\boldsymbol{p}$, $\boldsymbol{P}$ и $\boldsymbol{R}$», а к некоторому другому «миру $\boldsymbol{q}$, $\boldsymbol{Q}$ и $\boldsymbol{S}$». В случае справедливости такого предположения должно выполняться условие

$$\overline{\log K} \leqslant N(\boldsymbol{R}/\boldsymbol{S}) + \overline{N}(\boldsymbol{P}/\boldsymbol{Q}) - N(\overline{\boldsymbol{P}}/\overline{\boldsymbol{Q}}) + \overline{N}(\boldsymbol{p}/\boldsymbol{q}), \tag{65}$$

являющееся следствием формул (58) и (63).

Выполнение (65) не доказывает справедливости сделанного предположения*), однако нарушение (65) безусловно его опровергает. Удобством является то, что такое опровержение можно получить, не производя дополнительных опытов в «мире $\boldsymbol{q}$, $\boldsymbol{Q}$, $\boldsymbol{S}$».

*) Существует способ более сильной оценки средней неопределенности, заключающийся в построении по $\boldsymbol{q}$, $\boldsymbol{Q}$, $\boldsymbol{S}$ оптимальной матрицы $\| d'_{i,k} \|$, а по ней — $\| z \|$.

Высказанные соображения об оценке качества декодирующих алгоритмов применимы, разумеется, и к системам без жесткого порядка перебора (формулы (37) и (58)). Однако для них эксперимент по выяснению неопределенности окажется значительно более громоздким, так как потребуется большое число раз решать одну и ту же частную задачу после каждого декодирования сигнала*).

Имеется еще одно обстоятельство, которое заставляет думать, что оценка декодирующих устройств может принести пользу в первую очередь для систем с обратной связью. Оно рассмотрено в следующем параграфе.

§ 8. Трудность и неопределенность

Вернемся теперь к вопросу, который мог возникнуть еще в самом начале: почему в качестве характеристики трудности задачи для данного решающего алгоритма мы избрали именно логарифм среднего числа проб? Почему не число проб или не квадрат числа проб? Совершенно очевидно, что в разных конкретных случаях «цена» за применения W-алгоритма будет весьма различной. Иногда она будет просто пропорциональна числу применений, например, если реализация W-алгоритма — относительно самое длинное место в счетной программе, а ценим мы машинное время. В другом случае, если много времени и затрат уходит на создание экспериментальной установки, а сами эксперименты (применение W-алгоритма) «стоят» дешево, целесообразной оценкой трудности будет некоторая функция $F(k)$ такая, что $F(1) \gg F(k)-F(k-1) > 0$ при $k > 1$.

Универсального способа оценки трудности не может быть. Что же характеризуется «неопределенностью»?

Для ответа на этот вопрос вспомним, что для решения задачи система ведет экспериментальную работу. Система изначально «знает», что ответом является одно из b_i, но в нее не заложены сведения, какое именно b_i есть ответ данной частной задачи. Недостаток знаний решающий алгоритм восполняет с помощью опытов. Существует и другой способ получения сведений об ответе задачи. Это использование сообщений, приходящих по каналу связи. Естественно попытаться найти «коэффициент эквивалентности» между этими двумя способами получения сведений. В каких единицах нужно «измерять эксперимент», необходимый для решения задачи, чтобы уметь предсказывать изменения, которые могут возникнуть в этой величине после прихода данного сигнала?

Формулы типа (35) или (57) показывают, что связь между «количеством необходимого эксперимента» и свойствами канала связи и декодирующего устройства легко устанавливается, если измерять «необходимый эксперимент» неопределенностью задачи.

Может вызвать недоумение, казалось бы, искусственный и непонятный способ вычисления неопределенности — «среднее от логарифмов средних». Однако этот «искусственный» метод иногда приводит к результату, более соответствующему нашим интуитивным представлениям, чем, например, вычисление среднего числа проб.

Приведем пример такой ситуации. Пусть имеется задача, для решения которой алгоритм, исходящий из гипотезы о равномерном распределении вероятностей ответов, тратит в среднем 10^9 проб. Пусть, далее, мы имеем две системы, решающие эту задачу. Первая система обладает следующим свойством: в 99% случаев она решает задачу в среднем за 1000 проб, а в 1% случаев — за 10^9 проб. Вторая система в 99% случаев находит решение за 10 проб, а в 1% — за те же 10^9 проб.

*) Это, кстати, не всегда можно осуществить, так как нельзя заставить животное забыть результат предыдущего опыта.

На вопрос: «какая из систем лучше решает задачу?» — всякий человек ответит: «вторая система заметно лучше».

Попробуем сравнить работу систем, опираясь на среднее число проб. Для первой системы среднее число проб равно

$$0{,}99 \cdot 1000 + 0{,}01 \cdot 10^9 \approx 10^7 \cdot 1{,}0001.$$

Для второй системы

$$0{,}99 \cdot 10 + 0{,}01 \cdot 10^9 \approx 10^7 \cdot 1{,}000001.$$

Разница столь ничтожна, что, пользуясь критерием «среднее число проб», мы признали бы обе системы практически равноценными при решении этой задачи.

Сравним теперь системы, пользуясь неопределенностью задачи для них (будем брать десятичные логарифмы). Для первой системы неопределенность равна

$$0{,}99 \cdot \lg 1000 + 0{,}01 \cdot \lg 10^9 = 0{,}99 \cdot 3 + 0{,}01 \cdot 9 = 3{,}06.$$

Для второй системы

$$0{,}99 \lg 10 + 0{,}01 \lg 10^9 = 0{,}99 \cdot 1 + 0{,}01 \cdot 9 = 1{,}08.$$

Разница очень заметна и хорошо соответствует интуитивному ощущению, что вторая система «почти всегда» решает задачу в сто раз быстрее, чем первая, а в 1% случаев обе системы решить задачу просто не могут.

Мы нарочно рассмотрели случай весьма различного числа проб при решении системами разных частных задач. В случае, когда все частные задачи решаются каждой системой примерно за одинаковое число проб, оценки с помощью среднего числа проб не приходят в противоречие со здравым смыслом. Оценка с помощью неопределенности также дает в этом случае осмысленный результат (только выраженный в логарифмическом масштабе).

Итак, при сравнении двух систем, решающих некоторую задачу, среднее число проб и неопределенность либо дают согласные оценки, либо, если оценки расходятся, то ближе к интуитивной оценке оказывается неопределенность.

Может также возникнуть вопрос о целесообразности понятий «оптимальный решающий алгоритм» или «оптимальный декодирующий алгоритм». Действительно, оптимальный для неопределенности решающий алгоритм может быть совсем не оптимальным, например, для среднего количества применений W-алгоритма *).

Покажем, что, несмотря на такое расхождение, во многих случаях выяснение вопроса об оптимальности декодирующего алгоритма в отношении неопределенности может предрешить вопрос о его оптимальности и с других точек зрения, например в отношении числа применений W-алгоритма.

Введем понятие «трудность» задачи для данного решающего алгоритма. Пусть $F(x)$ — неубывающая функция, определенная при $x \geqslant 1$. *Трудностью* частной задачи a_j для данного решающего алгоритма будем называть $T(a_j) = F(\bar{K}_j)$, где $\bar{K}_j$ — среднее число применений W-алгоритма при решении a_j.

Трудностью задачи $\boldsymbol{A}$ назовем:

$$T(\boldsymbol{A}) = \sum_j p(a_j) T(a_j) = \sum_j p(a_j) F(\bar{K}_j). \tag{66}$$

*) Для алгоритмов без обратной связи при $n=2$ наименьшее число применений W-алгоритма достигается не при $q_1 = p_1$ и $q_2 = p_2$, а при

$$q_1 = \frac{p_1 - \sqrt{p_1 - p_1^2}}{2p_1 - 1}; \quad q_2 = \frac{p_2 - \sqrt{p_2 - p_2^2}}{2p_2 - 1}.$$

Будем называть F *функцией цены трудности*. Неопределенность есть трудность с логарифмической функцией цены. Назовем $F(x)$ *выпуклой*, если при $\alpha_1 \geqslant 0$, $\alpha_2 \geqslant 0$ и $\alpha_1 + \alpha_2 = 1$

$$\alpha_1 F(x_1) + \alpha_2 F(x_2) \leqslant F(\alpha_1 x_1 + \alpha_2 x_2). \tag{67}$$

Теорема. *При любой выпуклой функции цены, в случае* $\sum p_i = 1$, *наименьшей будет трудность для решающего алгоритма, проверяющего* b_i *в порядке убывания вероятностей* p_i.

Вначале заметим, что повторные проверки одних и тех же b_i могут лишь увеличить среднее число проб, а значит, и трудность. Поэтому мы рассматриваем только алгоритмы с обратной связью. Будем считать, что индексы при b_i расставлены так, что $p_1 \geqslant p_2 \geqslant \ldots \geqslant p_n$.

Обозначим через $\pi_{i,\,l}$ вероятность того, что *если* ответ b_i, то он будет найден на l-м такте. Очевидно,

$$\sum_{l=1}^{n} \pi_{i,\,l} = 1, \tag{68}$$

так как ответ обязательно будет найден не более чем за n тактов. Можно также показать, что

$$\sum_{i=1}^{n} \pi_{i,\,l} = 1. \tag{69}$$

Соотношение (69) имеет место благодаря тому, что $\pi_{i,\,l}$ является одновременно условной вероятностью проверки b_i на l-м такте, *если* первые $l-1$ тактов не обнаружили ответа.

Трудность задачи $\boldsymbol{A}$ для решающего алгоритма, характеризуемого матрицей $\|\pi_{i,\,l}\|$, будет:

$$T(\boldsymbol{A}) = \sum_i p_i F\Big(\sum_l l \cdot \pi_{i,\,l}\Big). \tag{70}$$

Обозначим $T(\boldsymbol{A})$ через $f(\boldsymbol{p}, \|\pi_{i,\,l}\|)$.

Введем функцию f' следующим образом:

$$f'(\boldsymbol{p}, \|\pi_{i,\,l}\|) = \sum_i p_i \sum_l \pi_{i,\,l} F(l). \tag{71}$$

Благодаря выпуклости F и равенству (68)

$$f(\boldsymbol{p}, \|\pi_{i,\,l}\|) \geqslant f'(\boldsymbol{p}, \|\pi_{i,\,l}\|). \tag{72}$$

Рассмотрим матрицу

$$\|\pi'_{i,\,l}\| = \left\| \begin{array}{cccccc} \pi_{1,1} & \ldots & \pi_{1,l_1} & \ldots & \pi_{1,l_2} & \ldots \pi_{1,n} \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ \pi_{i_1,1} & \ldots & \pi_{i_1,l_1} + \alpha & \ldots & \pi_{i_1,l_2} - \alpha & \ldots \pi_{i_1,n} \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ \pi_{i_2,1} & \ldots & \pi_{i_2,l_1} - \alpha & \ldots & \pi_{i_2,l_2} + \alpha & \ldots \pi_{i_2,n} \\ \ldots & \ldots & \ldots & \ldots & \ldots & \ldots \\ \pi_{n,1} & \ldots & \pi_{n,l_1} & \ldots & \pi_{n,l_2} & \ldots \pi_{n,n} \end{array} \right\|, \tag{73}$$

полученную из $\|\pi_{i,\,l}\|$ изменением четырех элементов, где $\alpha > 0$, $l_2 > l_1$ и $i_2 > i_1$.

При этом

$$f'(\boldsymbol{p}, \|\pi_{i,\,l}\|) - f'(\boldsymbol{p}, \|\pi'_{i,\,l}\|) = \alpha (p_{i_1} - p_{i_2}) [F(l_2) - F(l_1)] \geqslant 0. \tag{74}$$

Свойства (68) и (69) сохраняются после преобразования (73).

Пусть $\pi_{1,\,t}$ — первое отличное от $\pi_{1,\,1}$ и не равное нулю число в первой строке матрицы $\|\pi_{i,\,l}\|$, а $\pi_{s,\,1}$ — первое отличное от $\pi_{1,\,1}$ и не рав-

ное нулю число в первом столбце. Произведем преобразование (73), приняв $i_1 = 1$; $i_2 = s$; $l_1 = 1$; $l_2 = t$, а за α взяв меньшее из чисел $\pi_{1,t}$ и $\pi_{s,1}$. Если в исходной матрице $\|\pi_{i,l}\|$ элемент $\pi_{1,1}$ не был равен единице, то после преобразования в матрице $\|\pi'_{i,l}\|$ в первой строке или в первом столбце станет на один нуль больше.

Будем повторять эту операцию до тех пор, пока не получим матрицу вида

$$\|\pi^2_{i,l}\| = \begin{vmatrix} 1 & 0 & 0 & \dots & 0 \\ 0 & \pi^2_{2,2} & \pi^2_{2,3} & \dots & \pi^2_{2,n} \\ 0 & \pi^2_{3,2} & \pi^2_{3,3} & \dots & \pi^2_{3,n} \\ \dots & \dots & \dots & \dots & \dots \\ 0 & \pi^2_{n,2} & \pi^2_{n,3} & \dots & \pi^2_{n,n} \end{vmatrix}.$$

Для этого понадобится не более $2n-3$ операций. Так как при каждом преобразовании выполняется соотношение (74), то

$$f'(p, \|\pi_{i,l}\|) \geqslant f'(p, \|\pi^2_{i,l}\|).$$

Проделаем ту же серию операций со второй строкой и вторым столбцом и т. д. до получения единичной матрицы $\|\pi^n_{i,l}\|$. Очевидно,

$$f'(p, \|\pi_{i,l}\|) \geqslant f'(p, \|\pi^n_{i,l}\|). \tag{75}$$

Но $f'(p, \|\pi^n_{i,l}\|)$ совпадает с $f(p, \|\pi^n_{i,l}\|)$—трудностью задачи A для алгоритма, пробующего с вероятностью единица на первом шаге b_1, на втором b_2 и т. д. Сопоставив это с (75), (72) и (70), получаем:

$$T(A) \geqslant f(p, \|\pi^n_{i,l}\|) = \sum_i p_i F(i), \tag{76}$$

чем и доказана теорема *).

Так как логарифм является выпуклой функцией, то для него справедлива эта теорема. Отсюда следует оптимальность алгоритма с «жестким порядком перебора», использованная в § 7.

Доказанная теорема имеет еще одно важное следствие: для всех выпуклых функций цены трудности существует один и тот же оптимальный декодирующий алгоритм, совпадающий с декодирующим алгоритмом, оптимальным для неопределенности. Поэтому если, например, цена трудности есть число проб (а не логарифм числа проб), то нарушение неравенства (64) все равно сигнализирует о том, что для выбранной цены декодирующий алгоритм может быть улучшен (так как линейная функция выпукла).

Итак, во многих случаях можно не уточнять, какой функцией цены трудности мы интересуемся. (А иногда это и было бы весьма затруднительно сделать. Например, как узнать, какую именно функцию числа проб ценит крыса, выбираясь из лабиринта?)

В § 2 рассматривалось понятие «оптимальной» гипотезы. Его можно, естественно, распространить на случай нелогарифмических функций цены трудности. Несколько сложнее с понятием «гипотезы, содержащей только истину». В определение (7), кроме неопределенности, входит еще и энтропия распределения q. Что будет аналогом энтропии при других функциях цены трудности?

---

*) Для некоторых практических приложений целесообразно определять трудность частной задачи a_j через $F(K)$, а не через $F(\overline{K})$, как это сделано выше. В этом случае $T(A) = f'(p, \|\pi_{i,l}\|)$. Отсюда следует справедливость доказанной теоремы и для трудности, введенной таким способом. При этом отпадает требование выпуклости функции $F(x)$; необходимо только, чтобы она была неубывающей.

Пусть $F(x)$ — функция цены трудности. Обозначим через $H^F(p)$ минимум трудности задачи A для решающего алгоритма без обратной связи. Легко видеть, что по отношению к рассматриваемой трудности функция H^F играет ту же роль, что и энтропия по отношению к неопределенности. В частности, гиперплоскость, касательная в точке t к поверхности с уравнением $H^F(q)$ *), выражает трудность задач при гипотезе t (ср. рис. 1).

Поэтому, если мы в определениях (7) и (8) заменим $H(q)$ на $H^F(q)$, то теорема о совпадении оптимальной гипотезы с сильнейшей гипотезой, содержащей только истину, останется справедливой при оценке трудности с помощью функции F.

Например, как показал Н. Д. Нюберг (частное сообщение), при линейной функции цены трудности, $H^F(p)=\left(\sum_i \sqrt{p_i}\right)^2$. Пользуясь вместо энтропии этим выражением, мы можем говорить о «гипотезах, содержащих только истину» при оценке не по неопределенности, а по среднему числу проб.

§ 9. Обычная задача на этом языке

В §§ 1 и 3 были введены понятия: W-алгоритм, решающий алгоритм, декодирующий алгоритм. Как перевести на этот язык такое привычное всем понятие, как «условие задачи»?

Начнем со «школьной» арифметической задачи. «У двух мальчиков есть двенадцать книг. У Коли на две книги больше, чем у Пети. Сколько книг у Коли и сколько у Пети?».

Во-первых, *условие содержит W-алгоритм.* Действительно, в принципе можно найти ответ, генерируя случайным образом пары чисел и проверяя, удовлетворяют ли они условию.

Во-вторых, условие является сигналом, который после соответствующего декодирования может сократить неопределенность задачи. В этой второй роли условие содержит также и указание множества объектов (возможных ответов), по отношению к которым W-алгоритм является высказыванием. В данной задаче это — множество пар чисел.

Очень трудно заметить на самом себе или на другом взрослом человеке наличие у «условия» двух существенно разных функций. В явном виде оно проявляется при экспериментах с детьми.

Если предложить эту задачу ребенку лет восьми-девяти, уверенно считающему до ста и легко производящему сложение, вычитание и деление на два в пределах первых двух десятков, но не решавшему подобных задач, то он, как правило, *не пользуется условием как W-алгоритмом.* Он высказывает непроверенные предположения, предпочитая использовать в качестве W-алгоритма взрослого, давшего ему задачу. При этом, однако, легко заметить, что он пользуется условием для сокращения области поисков (хотя часто и не наилучшим образом). В нашем примере ребенок почти никогда не назовет чисел, больших 12. Очень многие дети пробуют только пары, дающие в сумме 12. Иногда бывают случаи, когда пробуются пары с разностью два. Если ребенок знает дроби, то он все-таки называет только целые числа (количество книг не бывает дробным!), в то же время, если изменить формулировку условия и сказать: «... двенадцать килограммов конфет...», то появятся дробные пробы.

Все это указывает на наличие процесса декодирования условия на стадии, когда в качестве W-алгоритма условие еще не используется.

---

*) $H^F(p)$ выпукла, так как через каждую точку этой поверхности можно провести гиперплоскость, не пересекающую $H^F(p)$.

«Правильно составленное» условие задачи из школьного задачника после оптимального декодирования уменьшает неопределенность до нуля. Это приводит к тому, что иногда в 3—4-м классе ребята умеют выдавать правильный ответ задачи, но не умеют (или во всяком случае не имеют привычки) его проверить. Возможно поэтому с трудом решаются «нестандартные» задачи, в которых оптимальное декодирование условия лишь снижает неопределенность до такой степени, что становится возможным перебор оставшихся вариантов.

С этой точки зрения при решении «стандартной» арифметической задачи решающий алгоритм не работает. После декодирования условия неопределенность падает до нуля и «решать» уже нечего. Однако интуитивно мы чувствуем, что «решать» что-то приходится. Это «что-то» есть, по-видимому, некоторая другая задача — *синтез декодирующего алгоритма*, — которая может иметь весьма большую неопределенность.

Рассмотрим теперь задачу совсем из другой области. Пусть ее «условие» звучит так: «найти (построить) устройство, которое, будучи помещенным не ближе 300 *м* от поверхности 2×2 *м*, создает на ней в ясную погоду ночью освещенность не менее 100 люкс».

В этой задаче условие также содержит прямое указание на W-алгоритм, с помощью которого можно узнать, является ли данное устройство ответом задачи. Использовать же условие для снижения неопределенности задачи уже гораздо труднее, чем в первом примере. Такого снижения мы достигаем путем обращения к справочникам, книгам по светотехнике и т. д. (Строго говоря, второй пример от первого отличается лишь степенью универсальности тех сведений, которые необходимы сверх условия задачи.) При этом нужно заметить, что уже указание области объектов, к которым применим W-алгоритм, в первом примере резко сокращает поиск. Там W-алгоритм можно применить только к парам чисел и просто невозможно испытывать, например, такие объекты, как «корова» или «красный цвет». Во втором же примере W-алгоритм может быть применен к чрезвычайно широкому классу объектов. В принципе с помощью W-алгоритма можно проверить, не является ли ответом такой объект: «мужчина-блондин, думающий о сверхновой звезде», причем само *условие* задачи вовсе не отвергает целесообразности этой проверки.

Итак, условие задачи должно обязательно содержать W-алгоритм. В противном случае не указан способ отличить ответ от неответа и мы говорим: «задача еще не поставлена». Свойства же условия как сигнала, способного сократить перебор, могут быть очень разными у разных задач и, естественно, зависят также от декодирующего алгоритма (того, «что мы знаем еще о данном вопросе»).

Заключение

«Классическая» теория информации в основном рассматривает взаимоотношения сигнала с каналом связи. Настоящая статья посвящена взаимоотношениям сигнала и получателя сообщения, который использует сигнал для решения некоторой задачи. Этим определяется круг вопросов, где может оказаться полезным рассмотренный в статье формализм.

В § 7 приводился пример задачи об оптимальности декодирующего алгоритма, в которой целесообразно пользоваться введенными понятиями.

Другим примером задачи, выигрывающей от рассмотрения «полезной информации», является, по-видимому, вопрос об избыточности языка (например, письменной речи). В самом деле, для разных людей (для специалиста и неспециалиста в данной области или для человека грамотного и малограмотного) один и тот же текст имеет *разную* избыточность. Для

человека, хорошо знающего грамматику, избыточность текста больше, так как он может исправить опечатку, которую не заметит человек малограмотный.

Отсюда прямо следует способ измерения избыточности *данного текста для данного наблюдателя*. Найдем сначала среднюю *полезную информацию*, которую получает наблюдатель от одной буквы текста.

Для этого проводим опыт, похожий на опыт Шеннона с отгадыванием букв [6], следующим образом: показываем наблюдателю кусочек текста из x букв и просим его назвать вероятности, с которыми он ожидает появления на $(x+1)$-м месте различных букв алфавита. Пусть q_1 есть указанная им вероятность для той буквы, которая *на самом деле* стоит на $(x+1)$-м месте. Повторяем опыт L раз, выбирая по жребию участок текста. Если q_i есть результат i-го опыта, то средняя полезная информация, содержащаяся в $(x+1)$-й букве текста, для этого наблюдателя, знающего предыдущие x букв, будет:

$$I_{\text{п}} = -\frac{1}{L}\sum_i \log q_i.$$

Отсюда избыточность

$$R = 1 - \frac{I_{\text{п}}}{\log n} = 1 + \frac{\sum \log q_i}{L \log n}, \tag{77}$$

где n — число букв алфавита.

Если, следуя Шеннону, считать, что испытуемый полностью использовал знание статистических связей в языке, то полученная избыточность (при $x \to \infty$) и есть величина, интересовавшая Шеннона. Если считать, что испытуемый учитывал не все связи, то мы получаем избыточность *для данного наблюдателя*.

Наконец, отметим, что полученное в § 5 обобщение понятия пропускной способности канала связи ($N(\overline{P}/\overline{Q}) - \overline{H}(P)$), возможно, окажется полезным при рассмотрении неэргодических источников сообщений. В самом деле, для таких источников гипотеза (Q), построенная на основании предыдущего поведения, не совпадает с действительным поведением (P), и обычная формула для эргодических источников сообщений ($H(\overline{P}) - \overline{H}(P)$) неприменима.

* *
*

Автор приносит искреннюю благодарность М. Н. Вайнцвайгу, Р. С. Гутеру, А. Н. Колмогорову, А. А. Ляпунову, Н. Д. Нюбергу, Я. Г. Синаю, М. С. Смирнову и А. М. Яглому, ознакомившимся с рукописью и высказавшим целый ряд советов и возражений, которые автор постарался принять во внимание.

ЛИТЕРАТУРА

[1] Харкевич А. А., О ценности информации, Сб. «Проблемы кибернетики», вып. 4, Физматгиз, 1960.
[2] Яглом А. М. и Яглом И. М., Вероятность и информация, М., 1960.
[3] Голдман С., Теория информации, ИЛ, 1957.
[4] Бонгард М. М., Моделирование процесса узнавания на цифровой счетной машине, Биофизика 6, 2, 1961.
[5] Бриллюэн Л., Наука и теория информации, М., 1960.
[6] Shannon C. E., Prediction and entropy of printed English, Bell System Tech. Journ. 30, 1951.

Поступило в редакцию 1 VII 1961