Список литературы

1. Ковалев М. М. Матроиды в дискретной оптимизации. Минск, 1987.
2. Ковалев М. М., Миланов П. Б. // Докл. АН БССР. 1980. Т. 24. № 9. С. 784.

Поступила в редакцию 13.03.87.

УДК 801.73:681.3

Н. К. РУБАШКО, И. В. СОВПЕЛЬ

АВТОМАТИЧЕСКИЙ КОНТРОЛЬ ТЕКСТОВ ЕСТЕСТВЕННЫХ ЯЗЫКОВ. II

В первой части работы [1] в виде условия (*) дано формальное определение основных типов текстовых ошибок, возникающих в линии ЕЯ коммуникации. В качестве начальных условий решаемой задачи АКД нами рассматриваются следующие:

задан эталонный словарь D;

в каждом слове $Y \in T_0$ возможна только одна, независимо от типа, ошибка;

абсолютно одна и та же ошибка для равных слов из T_0 возможна только один раз.

Для $Y_i \in T_1$, $i = \overline{1, n}$ обозначим: $X^{(i)} = \{X_\xi^{(i)}\}$, $\xi = \overline{1, r_i}$ — множество всех значимых слов языка L, попарно с Y_i удовлетворяющих условию (*), включая и Y_i, если Y_i — значимое слово.

Обозначим: $\lambda(Y_i)$ — частота Y_i в T_1.

Пусть Y_i, $Y_j \in T_1$. Число $r = |i - j| - 1$ назовем расстоянием между Y_i и Y_j в T_1, обозначим $r(Y_i, Y_j)$.

Общий алгоритм обнаружения и исправления ошибок для каждого слова $Y \in T_1$ может быть описан следующим образом.

Шаг 1. Начало.

Шаг 2. Если $Y_i \in D$, то перейти к следующему шагу, иначе — к шагу 5.

Шаг 3. Если $|X^{(i)}| = 1$, то перейти к следующему шагу, иначе — к шагу 12.

Шаг 4. Считать Y_i «правильным» словом и перейти к шагу 13.

Шаг 5. Если $\lambda(Y_i) = 1$, то считать Y_i ошибочным словом и перейти к следующему шагу, иначе — к шагу 10.

Шаг 6. Если $|X^{(i)}| = 0$, то перейти к следующему шагу, иначе — к шагу 8.

Шаг 7. Считать Y_i словом с «грубой» ошибкой и перейти к шагу 13.

Шаг 8. Если $|X^{(i)}| = 1$, то перейти к следующему шагу, иначе — к шагу 11.

Шаг 9. Положить $Y_i = X_1^{(i)}$ и перейти к шагу 13.

Шаг 10. Считать Y_i «новым и правильным» словом и перейти к шагу 13.

Шаг 11. Выбрать из $X^{(i)}$ такое $X_\xi^{(i)}$, которое удовлетворяет S_1 и S_2 и положить $Y_i = X_\xi^{(i)}$. Перейти к шагу 13.

Шаг 12. Если в T_1 не существует Y_j такое, что $r(Y_i, Y_j) \leqslant 4$ и Y_j было обработано как ошибочное слово, то перейти к шагу 11, иначе — к шагу 4.

Шаг 13. Конец.

Замечание. При реализации алгоритма переход к шагу 12 осуществляется после того, как обработаны по остальным условиям все слова $Y_j \in T_1$ такие, что $r(Y_i, Y_j) \leqslant 4$. Оказалось, что если ошибки составляют примерно 0,55 % общего числа слов, то около 40 % из них — ошибки, приводящие к значимым словам. Учитывая это и то, что ошибки

распределяются более или менее равномерно в T_1, сделано допущение, что расстояние между ошибочными словами не может быть меньше 4, и поэтому на шаге 12 при определенных условиях предпочтение отдается значимому слову из T_1, которое признается «правильным». В этом еще раз проявляется отличие общего цикла формирования теории в области искусственного интеллекта от классического [2]. В «быстром» режиме работы системы АКД, когда используется только морфологический уровень языка, шаг 11 не является автоматической процедурой, а реализуется в диалоге. Случай $Y_i \notin D$, $\lambda(Y_i) = 1$ и $|X^{(i)}| = 0$ идентифицируется системой как случай «отказа».

Основным недостатком указанного варианта системы, реализованного для английского и русского языков, как и существующих систем подобного типа (см., например, [3]), является многозначность решения задачи АКД и невозможность автоматически идентифицировать случай $Y = l(X)$, где $Y \neq X$ и есть значимое слово языка L [4].

Предполагаются два метода реализации процедуры автоматического установления соответствия $X_\xi^{(i)} = X^{(i)}$ синтаксису S_1 и семантике S_2 языка L.

Процедуральный метод. В этом случае последовательно используются два критерия: принадлежность $X_\xi^{(i)} \in X^{(i)}$ словарному обороту языка L (множество словарных оборотов задается как подмножество в D) и принадлежность $X_\xi^{(i)} \in X^{(i)}$ определенному типу сегмента [5], устанавливаемому на этапе автоматического анализа T_1. Сама процедура анализа строится на основании семантико-грамматической информации, заложенной в словаре системы АКД и правил вывода (ПВ) типа «если..., то...», для описания которых используются специальные КС- и НС-грамматики (подобный механизм используется в большинстве существующих экспертных систем). Таким образом, имеет место следующий общий П-алгоритм:

Шаг 1. Начало.

Шаг 2. Выяснить, существует ли в T_1 словарный оборот, содержащий $X_\xi^{(i)}$. Если да, то перейти к следующему шагу, иначе — к шагу 4.

Шаг 3. Если $X_\xi^{(i)} = Y_i$, то считать Y_i «правильным» словом, если нет — ошибочным и положить $Y_i = X_\xi^{(i)}$. Перейти к шагу 5.

Шаг 4. Выполнить процедуру анализа T_1, используя последовательно подстановку слов из множества $X^{(i)}$ вместо Y_i. В случае успешного завершения процедуры перейти к шагу 3, иначе — к следующему шагу.

Шаг 5. Конец.

П-алгоритм реализован для английского языка с помощью стратегии восходящего анализа. Объем массива ПВ — около 4 тыс. так называемых однозначных правил. При этом первый из указанных критериев оказывается достаточно сильным «фильтром», поскольку покрываемость T_1 словарными оборотами составляет до 22 %. Проблема снятия многозначности остается в ряде случаев, требующих полного семантического анализа. Кроме того, алгоритм, основанный на ПВ, очевидно, работает только с текстом, который «следует» этим правилам.

Декларативный метод. Информационную основу этого метода составляет словарь D, в котором для каждого $X_i \in D$ указано конечное множество его семантико-грамматических категорий (кодов) в соответствии с множествами S_1 и S_2 языка L. Кроме того, по результатам исследований достаточно большого по объему корпуса текстов известна частота $\lambda(k_i)$ каждого кода $k_i \in K$ и частота встречаемости $\lambda(k_i, k_j)$ любой пары кодов из K, где K — конечное множество попарно различных кодов слов в языке L. Наконец, на множествах D и K задана 4-значная функция вероятности $P(X_i, k_j)$, принимающая значения, которые тоже известны, из множества $\{0; 0{,}01; 0{,}1; 1\}$ и описывающая вероятность события «слово X_i имеет в языке L код k_j».

Используя словарь D, поставим в соответствие для $X^{(i)}$ множество кодов $K^{(i)} = \{k_p^{(i)}\}$, $p = \overline{1, q_i}$. В то время как первый метод основан на выводе семантико-грамматических свойств текста, настоящий использует вероятностный подход к анализу ЕЯ и исходит из следующего утверждения.

Утверждение 1. Однородная цепь Маркова с конечным числом состояний является математической моделью грамматики, порождающей ограниченный ЕЯ.

Прежде всего решается задача выбора наиболее вероятного кода для Y_i из множества $K^{(i)}$, а затем Y_i заменяется на то слово из $X^{(i)}$, которому принадлежит выбранный код. Критерием выбора кода является $\max\{P_r(k_p^{(i)})\}$, $p = \overline{1, q_i}$, где $P_r(k_p^{(i)})$ — относительная вероятность кода $k_{(p)}^t$, рассчитываемая по формуле: $P_r(k_p^{(i)}) = \dfrac{P_a(k_p^{(i)})}{\sum\limits_{j=1}^{q_i} P_a(k_j^{(i)})}$, где $P_a(k_j^{(i)})$ — абсолютная вероятность кода $k_j^{(i)}$, которая вычисляется из соотношения $P_a(k_p^{(i)}) = P_{-1} \cdot P_0 \cdot P_1$, P_0 — 4-значная функция вероятности для данного кода и соответствующего ему слова из $X^{(i)}$; P_{-1}, P_1 — соответственно «предшествующая» и «последующая» вероятности, определяемые как рекурсивные функции, учитывающие предшествующий и последующий для Y_i контекст.

Утверждение 2. Контекст для вычисления абсолютной вероятности кода Y_i определяется в T_1 такой цепочкой $Y_m Y_{m+1} \ldots \overline{Y_{i-1} Y_i Y_{i+1}} \ldots Y_{n-1} Y_n$, для которой $|K^{(m)}| = |K^{(n)}| = 1$ и $|K^{(j)}| > 1$ для $j = \overline{m+1, n-1}$.

Последовательность кодов, взятых по одному из множеств $K^{(m)}$, $K^{(m+1)}$, ..., $K^{(n)}$, назовем кодовой цепочкой для $k_j^{(i)}$. Если она содержит $k_j^{(i)}$, то назовем ее собственной цепочкой для данного кода. Величину $f_i = \dfrac{\lambda(\overline{k_i}, \overline{k_{i+1}})}{\lambda(\overline{k_i}) \cdot \lambda(\overline{k_{i+1}})}$, где $\overline{k_i}$, $\overline{k_{i+1}}$ — элементы кодовой цепочки, назовем функцией относительной вероятности кода $\overline{k_i}$.

Обозначим: m_1 — общее количество кодовых цепочек для $k_j^{(i)}$; n_1 — количество его собственных цепочек; l_1 — длина кодовой цепочки; $P_{s\eta}$ и $F_{s\eta}$ — соответственно 4-значная функция и функция относительной вероятности η-го элемента 3-й цепочки.

Утверждение 3.

$$P_r(k_p^{(i)}) = \frac{\sum\limits_{s=1}^{n_1} F_{s1} \prod\limits_{\eta=2}^{l_1-1} F_{s\eta} P_{s\eta}}{\sum\limits_{s=1}^{m_1} F_{s1} \prod\limits_{\eta=2}^{l_1-1} F_{s\eta} P_{s\eta}}.$$

Таким образом, имеет место следующий общий Д-алгоритм.

Шаг 1. Начало.

Шаг 2. Каждому значимому слову $Y_i \in T_1$ поставить в соответствие подмножество $K^{(i)} \in K$ (виртуальная модель T_1 на уровне слов [5]).

Шаг 3. Слову Y_i, обрабатываемому общим алгоритмом АКД, в случае перехода к шагу 11 поставить в соответствие подмножества $K_1^{(i)}$, $K_2^{(i)}$, ..., $K_{r_i}^{(i)}$.

Шаг 4. Вычислить относительную вероятность каждого кода из подмножеств $K_1^{(i)}$, $K_2^{(i)}$, ... $K_{r_i}^{(i)}$.

Шаг 5. Если Y_i значимое слово, то перейти к шагу 7.

Шаг 6. Считать Y_i ошибочным словом и положить $Y_i = X_k^{(i)} \in X^{(i)}$ такому, для которого относительная вероятность кода оказалась наибольшей. Перейти к шагу 8.

Шаг 7. Если относительная вероятность кода, соответствующего Y_i, оказалась наибольшей, то считать Y_i «правильным» словом и перейти к следующему шагу, иначе — к шагу 6.

Шаг 8. Конец.

Замечание. Основным достоинством Д-алгоритма является его сравнительная простота и высокая эффективность, а также возможность обнаруживать и исправлять «грубые» ошибки в T_1.

Список литературы

1. Рубашко Н. К., Совпель И. В. // Вестн. Белорус. ун-та. Сер. 1: Физ. Мат. Мех. 1989. № 1. С 48.
2. Klaus K. Obermeier, David de Hilster // Applications of Artificial Intelligence II. 1985. V. 548. P. 220.
3. Белоногов Г. Г., Штурман Я. П., Кузнецов Б. А., Поздняк М. В. // Вопросы информац. теории и практики. М., 1984. № 51. С. 24.
4. Atwell E. // ICAME NEWS. 1983. N 7. P. 13.
5. Совпель И. В. // Международный семинар по машинному переводу: Тез. докл. М., 1983 С. 205.

Поступила в редакцию 04.06.87.

УДК 519.1

В. Э. ЗВЕРОВИЧ, И. Э. ЗВЕРОВИЧ

О СВЯЗЫВАЮЩЕМ ЧИСЛЕ ГРАФОВ

Рассматриваются конечные неориентированные графы без петель и кратных ребер. Неопределяемые понятия имеются в [1]. Связывающим числом [2] графа G с множеством вершин VG называется величина $\mathrm{bind}(G)=\min|N(X)|/|X|$ по всем $X\in F=\{X/\varnothing\neq X\subseteq VG,\ N(X)\neq VG\}$, где $N(X)=\bigcup\limits_{x\in X} N(x)$, а $N(x)$ — множество вершин, смежных с x.

В [3] поставлена задача характеризации графов, для которых

$$\mathrm{bind}(G)=\min_{X\in I}|N(X)|/|X|, \tag{1}$$

где I — семейство всех независимых множеств графа G. В работе рассматривается случай $\mathrm{bind}(G)>1$, поскольку при $\mathrm{bind}(G)\leqslant 1$ результат известен.

Через n, δ, α обозначаются соответственно порядок, минимальная степень вершин и число независимости графа G, а через $<X>$ — подграф графа G, индуцированный множеством $X\subseteq VG$.

Утверждение 1. Если $\mathrm{bind}(G)=|N(X)|/|X|>1$ и граф $<X>$ имеет изолированную вершину, то $X\cup N(X)=VG$.

Доказательство. Допустим противное: $Z=VG\setminus(X\cup N(X))\neq\varnothing$. Образуем множество $Y=X\cup Z$. Очевидно, что между X и Z нет ребер, поэтому $N(Y)=N(X)\cup N(Z)\subseteq N(X)\cup Z$. Тогда

$$|N(X)|/|X|=\mathrm{bind}(G)\leqslant|N(Y)|/|Y|\leqslant(|N(X)|+|Z|)/(|X|+|Z|). \tag{2}$$

Заметим, что если $<X>$ не содержит изолированной вершины, то $N(Y)$, возможно, совпадает с VG, т. е. $Y\notin F$, и неравенством $\mathrm{bind}(G)\leqslant|N(Y)|/|Y|$ пользоваться нельзя.

Из (2) следует, что $|N(X)|\,|Z|\leqslant|X|\,|Z|$, а поскольку $Z\neq\varnothing$, то $|N(X)|\leqslant|X|$ — противоречие с условием.

Утверждение 2. Если $\mathrm{bind}(G)>1$, то равенство (1) равносильно условию $\mathrm{bind}(G)=n/\alpha-1$.

Доказательство. Пусть U — наибольшее независимое множество графа G. Если $\mathrm{bind}(G)=n/\alpha-1$, то $\mathrm{bind}(G)=|N(U)|/|U|$, т. е. справедливо (1). Пусть выполняется (1), т. е. существует $X\in I$, для ко-