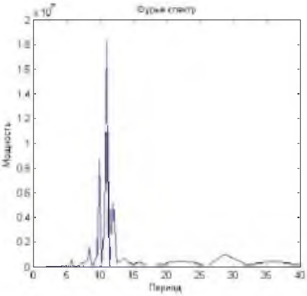

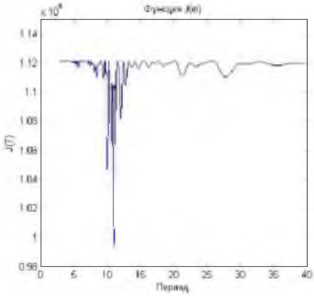

Вычислительная сложность задачи логического анализа для адаптивно-логической модели естественного языка

В.А. Крайванова

АлтГТУ, г. Барнаул

Человеко-машинный интерфейс на естественном языке (ЕЯ) не является универсальным, однако существует достаточно широкий круг задач, в которых человеку проще сформулировать свое пожелание в виде фразы на ЕЯ, чем в терминах формальных человеко-машинных интерфейсов. Причем эти задачи не требуют полноценного семантического анализа всех нюансов речи, зачастую достаточно буквального понимания фраз. Для создания ЕЯ-интерфейса предлагается модель логического анализа фраз.

Пусть W – конечное множество известных в модели слов, на котором введены отношение синонимичности и иерархия понятий. В качестве представления фраз на ЕЯ нами выбрана функциональная форма, функциональными символами являются слова из лексикона W, а типами аргумента синтаксическими отношениями между ними. О функциональной форме удобно говорить, как о дереве. Такое помеченное дерево фразы (ПДФ) в том или ином виде строят практически все синтаксические анализаторы. Задача логического анализа фразы заключается в том, чтобы свести некоторую начальную фразу ϕ к множеству фраз, определяющих команды и понятных объекту управления. Механизм преобразования фраз описывается конечным множеством правил контекстной замены *Rules*. В общем виде правило задается парой $r = \langle H \to C, Parameters \rangle$, где H и C – гипотеза и следствие правила, представленные двумя ПДФ, некоторые вершины которых, возможно, помечены не словами из W, а параметрами. *Parameters* – множество параметров правила. Значением параметра p_i может стать ПДФ, корень которого помечен словом, обобщаемым некоторым множеством понятий $N_i \subseteq W$. Часть правил являются заключительными и представляют собой шаблоны команд объекта управления. На основе введенных определений задачу логического анализа фразы ϕ можно сформулировать следующим образом: найти множество E_ϕ всех заключительных функциональных форм, выводимых из данной функциональной формы ϕ с помощью цепочек вывода длины не более *Lenght*.

Эта задача является NP-трудной. Действительно, пусть $T = <K, \Sigma, \delta, q_0, f. a_0>$ – машина Тьюринга (МТ), решающая некоторую произвольную NP задачу. Здесь K – множество состояний МТ, Σ – алфавит ленты МТ, $\delta=\{q_i a \rightarrow q_j bD \mid q_i, q_j \in K; a, b \in \Sigma, D \in \{R, L, E\}\}$ – функция переходов, R – смещение головки вправо, L – смещение головки влево, E - головка стоит на месте; q_0 – начальное состояние МТ; $f.$– конечное состояние МТ; a_0 – символ пустой ячейки. Сведем задачу, решаемую машиной T к Задаче 1 следующим образом:

Множество слов $W=K\cup\Sigma\cup\{SYMBOL, SPECWORD\}$, $SYMBOL$ – понятие, обобщающее все слова из множества Σ, $SPECWORD$ - специальное слово для обозначения начала и конца текущей цепочки.

Конфигурация МТ $<a_1,...a_n, q_i, b_1,...b_m>$ представляется в виде функциональной формы следующим образом: $w_0(a_1(\,..(a_n(q_i(b_1(\,..(b_m(w_0))...)$. Множество правил *Rules* получается из функции переходов δ следующим образом.

Команда вида $q_i a \rightarrow q_j bE$ описывается правилом

$< q_i(a(\$X)) \rightarrow q_j(b(\$X)),\ Parameters = \{<\$X, \{SYMBOL\}>\} >.$

Команда вида $q_i a \rightarrow q_j bR$ описывается правилами

$< q_i(a(\$X)) \rightarrow b(q_j(\$X)),\ Parameters = \{<\$X, \{SYMBOL\}>\} >,$

$< q_i(a(SPECWORD)) \rightarrow b(q_j(w_0(SPECWORD))),\ Parameters = \varnothing >.$

Команда вида $q_i a \rightarrow q_j bL$ описывается правилами

$< \$Y(q_i(a(\$X)) \rightarrow q_j(\$Y(b(\$X))),\ Parameters = \{<\$X, \{SYMBOL, SPECWORD\}>, < \$Y, \{SYMBOL\}> \} >,$

$< SPECWORD\ (q_i\ (a\ (\$X))) \rightarrow SPECWORD\ (q_j\ (w_0\ (b\ (\$X)))),\ Parameters = \{<\$X, \{SYMBOL, SPECWORD\}> \}.$

Заключительное правило будет иметь следующий вид: $f.$

Очевидно, что применение правила к функциональной форме конфигурации МТ равносильно выполнению команды над конфигурацией. Таким образом, любая NP задача сводится к рассматриваемой задаче, а значит, рассматриваемая задача является NP-трудной.

Чтобы построить достаточно эффективные для практического использования алгоритмы, разработан ряд оптимизирующих эвристик на основе структуры ПДФ и специальная структура данных на основе схем ПДФ. Схема *Template*(ϕ) – это дерево фразы ϕ без пометок вершин. Модель дополнена адаптивными механизмами, позволяющими извлекать новые знания для лексикона и базы правил из предложений на ЕЯ непосредственно в процессе функционирования модели.

Предложенная математическая модель реализована в виде ядра программной архитектуры AL System для создания человеко-машинных интерфейсов на естественном языке (Свидетельство о государственной регистрации программы для ЭВМ №2009610579).

Внутренний интервал для множества решений системы линейных неравенств

И.А. Шарая

ИВТ СО РАН, Новосибирск

Рассмотрим систему линейных неравенств $Ax \geq b$, где A – заданная вещественная матрица размера m на n без нулевых строк, b – заданный вещественный вектор длины m, x – неизвестный вещественный вектор длины n. Обозначим множество решений этой системы через S, а i-ую строку матрицы A через $A_{i:}$.

Для формулировки результатов нам понадобятся некоторые термины из интервального анализа. Приведем их в геометрическом толковании.

Интервалом на вещественной оси называется всякий отрезок или точка. Если на каждой координатной оси в R_n мы выберем по интервалу и возьмем их прямое произведение, получим множество, которое называется интервальным вектором или интервалом в R_n.

Рассмотрим интервал в R_n. Нижним его концом называется точка этого интервала, имеющая минимальные координаты по всем компонентам, а верхним концом – точка, имеющая максимальные координаты по всем компонентам. Шириной интервала называется разность верхнего и нижнего концов. Радиусом интервала называется половина его ширины. Центром (серединой) интервала называется полусумма концов.

Интервалы, в отличие от вещественных чисел и векторов, обозначаются жирным курсивом.

Перейдем к оцениванию множества решений системы линейных неравенств.

Интервал $\boldsymbol{x}$ из R^n будем называть внутренним интервалом для множества S, если $\boldsymbol{x}$ содержится в S.

Утверждение 1. *Интервал $\boldsymbol{x}$ является внутренним для множества S тогда и только тогда, когда $A\boldsymbol{x} \geq b$ в интервальной арифметике.*