

Бісікало О.В. (Україна, Вінниця)

## Система «питання–відповідь» в межах моделі образного мислення

Відсутність повноцінного натурально мовного діалогу користувача з програмою є однією з головних причин, що гальмує впровадження експертних систем в освітянській сфері. Відомі системи «питання-відповідь», що працюють за сценарним принципом, мають обмежене використання внаслідок унікального характеру сценарію для кожного класу бази знань. Побудова аналогічних предикативних систем потребує трудомісткої логічної формалізації всіх вихідних положень, причому це можливо досягти тільки для вузького кола предметних галузей. Тому актуальною можна вважати задачу побудови системи «питання–відповідь», база знань якої ґрунтується на множині текстів, що характеризують обрану предметну галузь.

З метою розв'язання поставленої задачі пропонується використати модель образного мислення, в основу якої закладено наступну формальну ієрархію понять:

- **Образ** – множина  $Image = \{OQ, O, N, M, MQ\}$ , в якій виділяють п'ять підмножин–складових образа:  $OQ$  – якість об'єкту;  $O$  – об'єкт;  $N$  – поняття;  $M$  – метод;  $MQ$  – якість методу;
- **Асоціативний зв'язок** між образами – множина  $Links - syntagma = \{1, 2, 3, 4, 5, 6, 7\}$ , де цифрами позначені основні типи асоціативних зв'язків у вигляді підмножин питальних займенників, а саме: 1 – визначення; 2 – присудок; 3 – підмет; 4 – обставина місця; 5 – обставина часу; 6 – обставина; 7 – додаток;
- **Асоціативна пара** образів – формується у вигляді списку  $A - tw_j = (M_j(X_j S_j))$ , де  $M$  і  $S$  – головний та підлеглий образи, а  $X$  – елемент множини зв'язків;
- **Синтагма** – операція  $e_i$  деревоподібного зв'язування асоціативних пар, що виражає образне представлення події і є аналогом простого речення;
- **Текст** – операція  $Text$  об'єднання синтагм в зв'язну підмножину, в якій кожна з синтагм має з усіма іншими хоча б один спільний образ;
- **Довготермінова пам'ять** – операція  $Long-memory$  об'єднання синтагм всіх текстів в єдину множину, що моделює часову послідовність (стрічку) подій;
- **Асоціативна пам'ять** – операція  $Assoc-memory$  перетину асоціативних пар з різних синтагм довготермінової пам'яті або окремих текстів у вигляді ускладненої матриці інцидентності, в якій кожному образу відповідає відсортований підсписок асоційованих з ним образів;

Модель образного мислення представимо у вигляді двохосновної алгебраїчної системи  $Algebra = \langle B; \Omega \rangle$ , де  $B = \{Image, Links - syntagma\}$  - основи, а  $\Omega = \{IF, OP\}$  - сигнатура системи, причому  $IF$  – предикати, а  $OP$  - операції. В свою чергу, позначивши  $Add-OP$  додаткові операції алгебраїчної системи, маємо  $OP = \{A - tw, e, Text, Long - memory, Assoc - memory, Add - OP\}$ . В рамках запропонованого підходу можна формально визначити наступні функції:

- **Асоціативний пошук** – функція  $Find-image$  за допомогою асоціативної пам'яті  $Assoc-memory$  ставить у відповідність одному вхідному образу відсортований список вихідних образів, причому застосовується подвійне сортування по двом ознакам – кількості повторювань асоціативної пари та часу виникнення події (порядковому номеру синтагми):

$$Find - image := \{Im_i, Assoc - memory \rightarrow Im_j\};$$

- **Відповідь на питання** по тексту – на вхід функції  $Answer-image$  подається частина синтагми з головним образом та питальним займенником від нього, а на виході визначається підлеглий образ асоціативної пари, знайденої в результаті застосування функції  $Find-image$ , з такої синтагми окремого тексту або довготермінової пам'яті в цілому, яка є найбільш близькою за структурою графу до вхідної синтагми:

$$Answer - image := \{e_i, Im_i, Links - syntagma \rightarrow Find - image \rightarrow Im_j, e_j\}.$$

Підхід, що пропонується, дозволяє експертній системі будувати базу знань на основі використання текстових файлів та забезпечує генерацію відповіді на питання користувача з посиланням на ті речення тексту, де цю відповідь було знайдено. Діючий прототип системи «питання–відповідь» в межах моделі образного мислення реалізовано у вигляді функцій відомої оболонки LISP-програмування DrScheme.