

УДК 004.81:378.147

КЛАСИФІКАЦІЯ ОБРАЗНОГО ПОШУКУ ТА МОДЕЛЮВАННЯ ІНСАЙТ

О.В. Бісикало

Вінницький державний аграрний університет, м. Вінниця

На основі базових концептів образу, асоціативної пари образів та синтагми пропонується класифікація образного пошуку. Як демонстрацію можливостей підходу до накопичення синтагматичних відношень між образами формалізовано алгоритм моделювання відомого психологічного феномена інсайту як одного з варіантів образного пошуку.

Пошук образів за допомогою асоціацій вважається основою природних інтелектуальних систем [1]. До сфер використання асоціативного пошуку в натурально-мовних конструкціях можна віднести експертні системи, бази даних та бази знань, інформаційно-пошукові системи, автоматичний переклад, програмно-педагогічні засоби тощо.

Наукова проблема лексичного процесора полягає в тому, що класична логіка, символічна парадигма та традиційні підходи інженерії знань показали свою обмеженість в досягненні глибини семантичного аналізу тексту. Феноменологічний характер відповідних психофізіологічних процесів людини вимагає нових підходів до моделювання [2]. Окремі аспекти образного мислення були предметом дослідження різних наук. До найбільш помітних результатів можна віднести публікації з психолінгвістики (А.А. Залевська), нейропсихології (А.Р. Лурія), трансформаційної породжуючої граматики (Н. Хомський), комп'ютерної лінгвістики (А.В. Анісімов), діалогових систем (П.І. Соснін).

Невирішеними частинами проблеми варто вважати [3]:

- 1 Побудову цілісної моделі образного мислення.
- 2 Дослідження в рамках моделі психологічно значущих феноменів.
- 3 Класифікацію варіантів образного пошуку.

Характерно, що кілька базових варіантів пошуку спостерігаються частіше в інтелектуальній діяльності, оскільки для їх реалізації потрібні лише синтагматичні відношення між образами. Всі інші варіанти образного пошуку можна вважати вторинними, похідними від базового типу внаслідок появи парадигматичних відношень (нових асоціативних зв'язків) між образами. Ідея підходу до створення простору асоціативних пар образів на основі інформації з подій, що відбуваються:

- визначити складовими пошуку образ, асоціативну пару образів та синтагму;
- закласти пріоритет синтагматичного мислення;
- застосувати принцип «навчання дитини» для накопичення парадигматичних зв'язків між образами;
- отримати алгебраїчну модель образного мислення.

Поставлення задачі: визначити ознаки базового типу пошуку, для реалізації якого потрібні лише синтагматичні відношення між образами; показати, що похідний від базового тип пошуку з'являється внаслідок

накопичення парадигматичних відношень між образами; отримати як демонстрацію підходу алгебраїчну модель такого психологічного феномена, як інсайт.

Розглянемо можливі варіанти образного пошуку, одиницями якого можна вважати образ, асоціативну пару та синтагму. Формально будемо вважати образами множину $x \in X$, асоціативними парами – множину пар $Y = \{x, a\}$, де a – сила асоціативного зв'язку між образами, що встановлюється відображенням $A : X \times X \rightarrow a$. Синтагми, або сукупність взаємопов'язаних у вигляді подій асоціативних пар, позначимо як множину $z \in Z$. Виходячи з цього, пропонується така класифікація образного пошуку (базовий тип позначено сірим кольором):

1 **Образ \rightarrow образ** у вигляді відсортованого за зменшенням списку найбільш близьких за силою асоціативного зв'язку образів:

$$\text{Hash - Table} : X \rightarrow X . \quad (1)$$

2 **Образ \rightarrow образ**, де обидва образи близькі синонімічно, наприклад, «світлий \leftrightarrow ясний», «реве \leftrightarrow стогне», «високий \leftrightarrow стрункий»:

$$\text{Syn}XX : X \rightarrow X . \quad (2)$$

3 **Образ \rightarrow пара**, в якій однією зі складових знайденої пари є даний образ («Дніпро широкий», «хвилі Дніпра»):

$$\text{Part}XY : X \rightarrow Y . \quad (3)$$

4 **Образ \rightarrow пара** – пошук синонімічного словосполучення до даного образу, наприклад, «обманувать \leftrightarrow водить за нос», «бездельничать \leftrightarrow бить баклуши»:

$$\text{Syn}XY : X \rightarrow Y . \quad (7)$$

5 **Образ \rightarrow синтагма**, коли по одному образу можна згадати речення, подію або прислів'я («гривонька \rightarrow ой чий то кінь стоїть»):

$$\text{Part}XZ : X \rightarrow Z . \quad (8)$$

6 **Пара \rightarrow образ** – результат впливу (влиття за Виготським) сенсу пари в образ, наприклад, «тело жирное \rightarrow пингвин», «гордо реет \rightarrow буревестник»:

$$\text{Insert}YX : Y \rightarrow X . \quad (9)$$

7 **Образ \rightarrow X-Image \rightarrow образ** – пошук невідомої ланки в ланцюгу образів, що інколи сприймається людиною як інсайт («Дніпро \rightarrow (широкий) \rightarrow світ»):

$$\text{Find} - 1 : X \times X \rightarrow X . \quad (4)$$

8 **Пара \rightarrow пара**, коли дві пари є синонімічними, наприклад, «мы победили \leftrightarrow враг бежит» або «красна девица \leftrightarrow писанная красавица»:

$$\text{Syn}YY : Y \rightarrow Y . \quad (10)$$

9 **Образ \rightarrow X-Image-1 \rightarrow X-Image-2 \rightarrow образ** – пошук двох невідомих ланок у ланцюгу образів, наприклад, «розплетена \rightarrow (бо з милим я) \rightarrow розлучена»:

$$\text{Find} - 2 : X \times X \rightarrow X \times X . \quad (5)$$

10 Пара \rightarrow синтагма, де пара є однією з складових синтагми («кот учений \rightarrow и днем и ночью все ходит по цепи кругом»):

$$PartYZ : Y \rightarrow Z . \quad (11)$$

11 Образ \rightarrow X-Image-1 \rightarrow ... \rightarrow X-Image-n \rightarrow образ – пошук n невідомих ланок у ланцюгу образів, який найчастіше використовується для побудови відповідей або висловлювань, наприклад, «Еней \rightarrow (був парубок моторний і хлопець хоч куди) \rightarrow козак»:

$$Find - n : X \times X \rightarrow \underbrace{X \times \dots \times X}_n . \quad (6)$$

12 Пара \rightarrow синтагма – пошук синонімічної синтагми до даної пари, наприклад, «стражи порядку \leftrightarrow моя милиция меня бережет»:

$$SynYZ : Y \rightarrow Z . \quad (12)$$

13 Синтагма \rightarrow образ – результат впливу (влиття) сенсу синтагми в образ, наприклад, «рука миллионопалая, сжатая в один громающий кулак \rightarrow партия»:

$$InsertZX : Z \rightarrow X . \quad (13)$$

14 Синтагма \rightarrow пара – результат впливу (влиття) сенсу синтагми в пару, наприклад, «и только небо тебя поманит \rightarrow синяя птица»:

$$InsertZY : Z \rightarrow Y . \quad (14)$$

15 Синтагма \rightarrow синтагма як дві синонімічні конструкції наприклад, «как с гуся вода \leftrightarrow Васька слушает да ест»:

$$SynZZ - 3 : Z \rightarrow Z . \quad (15)$$

Узагальнену класифікацію з 15 розглянутих варіантів образного пошуку наведено в таблиці 1 з урахуванням того, що три тривіальні випадки, коли результатом образного пошуку є складові частини більш об'ємної первинної образної конструкції, а також рідко вживані випадки не розглядалися.

Таблиця 1- Узагальнена класифікація образного пошуку

$\rightarrow \setminus \uparrow$	Образ	Асоціативна пара	Синтагма
Образ	1, 2	3, 4	5
Асоціативна пара	6, 7	8, 9	10, 11, 12
Синтагма	13	14	15

З метою моделювання інсайту пропонується алгоритм, що відповідає базовому типу образного пошуку № 7 узагальненої класифікації (табл.1). Будемо вважати, що i -й елемент множини образів X визначається двійковим кодом $Bi-I_i$. Тоді алгоритм можна формально подати у вигляді операції $Find-1(Bi-I_1, Bi-I_2)$, що знаходить такий образ, для якого сумарна сила асоціативних зв'язків з образами $Bi-I_1$ та $Bi-I_2$ є найбільшою з усіх можливих. Ідея алгоритму полягає у визначенні множини потенційно можливих з'єднувальних образів для пари $(Bi-I_1, Bi-I_2)$ та побудові для цієї множини власної хеш-таблиці $ListI$ за критерієм максимуму сумарної сили асоціативного зв'язку. Тим самим

вибір можливих варіантів образів для інсайту буде тим багатшим, чим більший простір асоціативних пар закладено в систему.

Для розв'язання задачі застосуємо базовий варіант образного пошуку *Hash-Table* (№ 1 згідно з класифікацією з табл.1). У результаті виконання операції *Hash-Table*($Bi-I$) до кожного з образів пари ($Bi-I_1$, $Bi-I_2$) отримаємо два відсортованих за зменшенням списки асоційованих образів *LeftList* та *RightList*, а задача зводиться до знаходження таких образів, які є в обох списках (перетину двох множин). Відомо, що знаходження спільних елементів у двох списках розмірністю n та m може бути досягнуте повним перебором усіх елементів ($n \times m$). Але врахування попередньої відсортованості списків дозволяє зменшити обсяг обчислень за допомогою введення штучних верхніх та нижніх обмежувачів кожного списку (рис. 1), позначених відповідно як *LeftTop*, *LeftBottom*, *RightTop*, *RightBottom*.

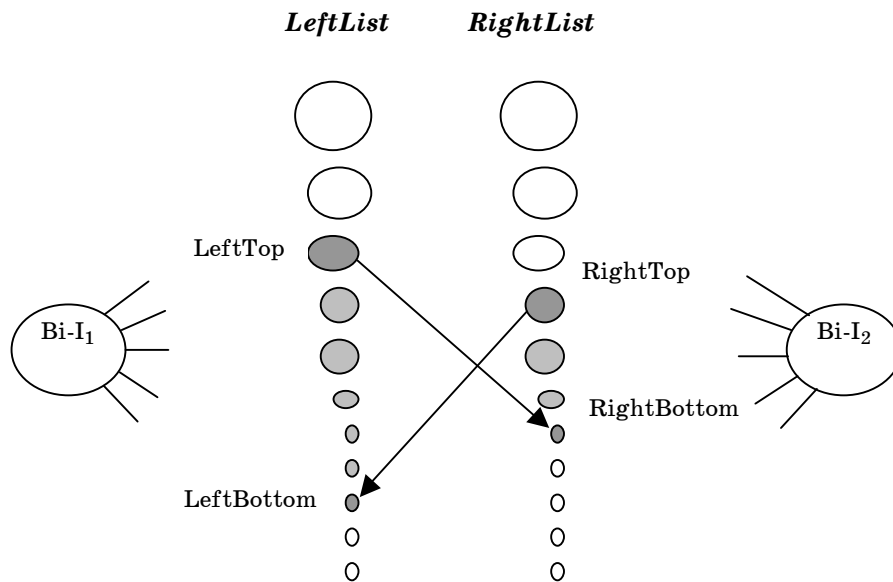


Рисунок 1- Верхні та нижні обмежувачі при пошуку спільних елементів двох списків

Позначимо двійкові коди та ваги образів у списках *LeftList* та *RightList* відповідно як $\langle BiL(i), Fl(i) \rangle$ та $\langle BiR(i), Fr(i) \rangle$, а кінцевого списку *ListI* як $\langle Bi-I(k), Fo(k) \rangle$, де i, j, k – порядкові номери (індекси) елементів у списках. Задача може бути розв'язана за допомогою такого узагальненого алгоритму:

1 а) Фіксуємо перший елемент списку *LeftList* та перебираємо список *RightList* до тих пір, поки в ньому не буде знайдено такий самий елемент. У цьому випадку номер елемента списку *LeftList* присвоюється змінній *LeftTop*, а номер елемента списку *RightList* – змінній *RightBottom*, далі формуємо код і вагу першого елемента списку *ListI* та переходимо до кроку 2.

б) Інакше – переходимо на наступний елемент списку *LeftList* та знову повертаємося до кроку 1а. Якщо список *LeftList* вичерпано, то видається повідомлення про те, що спільних елементів у двох списках не знайдено (після найскладнішого варіанта – повного перебору розмірністю $n \times m$).

2 а) Фіксуємо перший елемент списку *RightList* та перебираємо список *LeftList* з номера *LeftTop+1* до тих пір, поки в ньому не буде знайдено такий самий елемент. У цьому випадку номер елемента списку *RightList* присвоюється змінній *RightTop*, а номер елемента списку *LeftList* – змінній *LeftBottom*, далі формуємо код і вагу другого елемента списку *ListI* та переходимо до кроку 3.

б) Інакше – переходимо на наступний елемент списку *RightList* та знову повертаємося до кроку 2. Якщо список *RightList* дійшов до елемента *RightBottom*, то алгоритм закінчується і видається єдиний вже знайдений варіант спільного елемента *Bi-I(LeftTop)*.

3 Проводимо повний перебір елементів списків *LeftList* та *RightList* відповідно між парами елементів *LeftTop+1* та *LeftBottom-1*, *RightTop+1* та *RightBottom-1*, при знаходженні спільних елементів формуються *k*-ті складові списку *ListI* (починаючи з *k=3*).

Обґрунтування методу обмежувачів: потенційних елементів, більших за верхні обмежувачі, не існує внаслідок уже проведеної перевірки, а елементи, менші за нижні обмежувачі, перевіряти немає сенсу, оскільки їх сумарна вага не може бути більшою за вже знайдений варіант. При знаходженні елементів з усіма чотирма мітками розмірність пошуку обмежується зверху значенням

$$\begin{aligned} & \text{LeftTop} \times m + (n - \text{LeftTop}) \times \text{RightTop} + \\ & + (\text{LeftBottom} - \text{LeftTop} - 1) \times (\text{RightBottom} - \text{RightTop} - 1) \leq n \times m, \end{aligned} \quad (16)$$

причому два прийнятних варіанти пошуку перед третім перебором уже існують. Формально зобразимо алгоритм аналітичними виразами та граф-схемами з нотацією [4] по кроках:

$$\begin{aligned} \text{Step-1} ::= & \{ \overline{\text{LeftList} : i = 1, n} \} \{ \overline{\text{RightList} : j = 1, m} \} \{ \text{BiL}(i) = \text{BiR}(j) \} (i \rightarrow \text{LeftTop} * \\ & \text{BiL}(i) \rightarrow \text{Bi} - \text{I}(1) * \text{Fl}(i) + \text{Fr}(j) \rightarrow \text{Fo}(1) * j \rightarrow \text{RightBottom}), \text{Abort} \} \}, \end{aligned} \quad (17)$$

де *Step-1* – перший крок алгоритму; *Abort* – повідомлення про те, що розв’язку не існує;

$$\begin{aligned} \text{Step-2} ::= & \{ \overline{\text{RightList} : j = 1, \text{RightBottom}} \} \{ \overline{\text{LeftList} : i = \text{LeftTop} + 1, n} \} \\ & \{ \text{BiL}(i) = \text{BiR}(j) \} (j \rightarrow \text{RightTop} * \text{BiR}(j) \rightarrow \text{Bi} - \text{I}(2) * \\ & \text{Fl}(i) + \text{Fr}(j) \rightarrow \text{Fo}(2) * i \rightarrow \text{LeftBottom}), \text{Only} \} \}, \end{aligned} \quad (18)$$

де *Step-2* – другий крок алгоритму; *Only* – повідомлення про єдиний розв’язок *LeftTop*;

$$\begin{aligned} \text{Step-3} ::= & 2 \rightarrow k * \{ \overline{\text{LeftList} : i = \text{LeftTop} + 1, \text{LeftBottom} - 1} \} \\ & \{ \overline{\text{RightList} : j = \text{RightTop} + 1, \text{RightBottom} - 1} \} \{ \text{BiL}(i) = \text{BiR}(j) \} \\ & (k + 1 \rightarrow k * \text{BiL}(i) \rightarrow \text{Bi} - \text{I}(k) * \text{Fl}(i) + \text{Fr}(j) \rightarrow \text{Fo}(k), E) \}, \end{aligned} \quad (19)$$

де *Step-3* – третій крок алгоритму; *E* – пустий оператор.

У граф-схемах першого (рис.2), другого (рис.3) та третього (рис.4) кроків використані такі позначення для операторів структурного програмування:

- Do – цикл по параметру;
- If (+ -) – альтернатива;
- * – композиція.

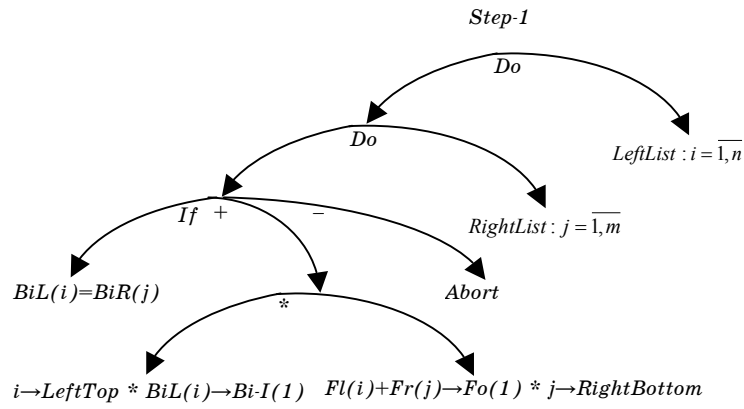


Рисунок 2 - Граф-схема кроку 1 алгоритму за методом обмежувачів

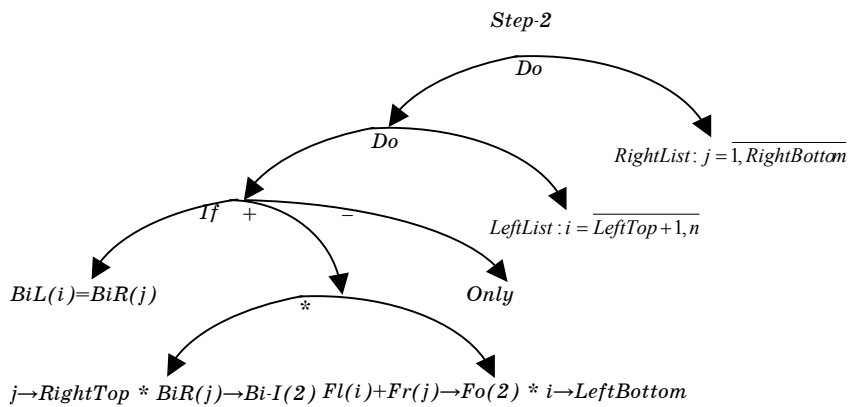


Рисунок 3 - Граф-схема кроку 2 алгоритму за методом обмежувачів

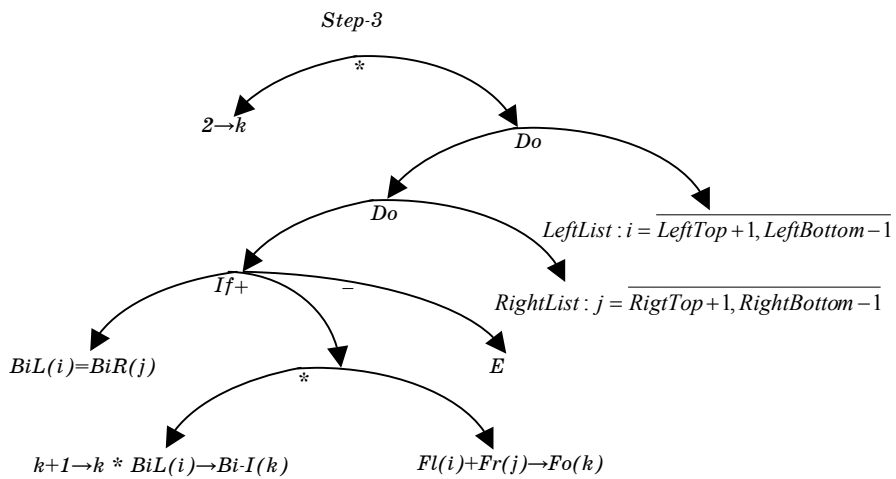


Рисунок 4 - Граф-схема кроку 3 алгоритму за методом обмежувачів

Кінцеве формування хеш-таблиці інсайтних образів досягається сортуванням списку $ListI$ за вагою $Fo(k)$, наприклад, за допомогою бульбашкового методу [4]. Алгоритм моделювання інсайту за методом обмежувачів програмно реалізовано на основі технології DrScheme + SQLite, яка поєднує можливості мов Lisp та SQL.

Таким чином, з п'ятнадцяти варіантів пошуку для можливих сполучень 3 основних образних конструкцій 5 варіантів можна віднести до базового синтагматичного типу, а 10 – до похідного парадигматичного типу. Запропонована класифікація дозволяє врахувати особливості парадигматичного мислення при безумовному пріоритеті синтагматичного, що відповідає онтогенезу розвитку мовлення.

Алгоритм моделювання інсайту за методом обмежувачів демонструє можливість спільного застосування варіантів образного пошуку № 1 та № 7 і дозволяє скоротити кількість варіантів перебору у випадку існування розв'язку задачі.

SUMMARY

THE CLASSIFICATION OF IMAGE SEARCH AND DESIGNING OF INSIDE

Bisicalo O.V.

Vinnitsa State Agrarian University

The classification of image search is offered on the basis of main concepts of image, associative pair of images and syntagma. As a demonstration of possibilities of the approach to the accumulation of syntagma relations between images the algorithm of designing of the known psychological phenomenon of inside is developed as one of the variants of image search.

СПИСОК ЛІТЕРАТУРИ

1. Гаврилов А.В. Архитектура программного обеспечения для поиска документов по запросу на естественном языке // Труды межд. конф. KDS-2001 "Знание-Диалог-Решение". – СПб. – 2001. – Т.1/ – С.124-130.
2. Бисикало О.В. Принципы построения концептуальной модели образного мышления // Первая международная конференция «Новые информационные технологии в образовании для всех», расширенные материалы конференции (29-31 мая 2006 г.). – Киев, 2006. – С. 25-34.
3. Бісікало О.В. Структура блоку пам'яті на основі моделі образного мислення людини // Искусственный интеллект – 2007. – № 3. – С. 461-468.
4. Цейтлин Г.Е. Введение в алгоритмику. – Киев: Издательство "Сфера", 1998.– 310 с.

Бісікало О.В., канд. техн. наук, доцент

Надійшла до редакції 23 квітня 2008 р.