УДК 004.2

МОЛЧАНОВСЬКИЙ О.І.,
ЗНАХУРЕНКО В.П.

МЕТОД ІНДУКТИВНОГО НАВЧАННЯ В ОСНОВІ РЕКОМЕНДАЦІЙНОЇ СИСТЕМИ ПОДАРУНКІВ

В цій статті розглянута задача надання рекомендацій щодо вибору подарунків. Метод розв'язку даної задачі відноситься до методів індуктивного навчання. Розроблено метод, який будує нечітке дерево прийняття рішень, використовуючи алгоритм FuzzyID3. Запропоновано метод надання рекомендації подарунків з фактором впевненості в кожній рекомендації.

The subject of the article is the task of making recommendations on the choice of gifts. Method for solving this problem is one of the methods of inductive learning. The method was developed for generating a fuzzy decision tree by using the algorithm FuzzyID3. The method that recommends some gifts with membership value of each one was proposed.

1. Вступ

Кожна людина стикалася з проблемою вибору подарунків. Для того, щоб допомогти людям, можна запропонувати систему, яка б за певними критеріями та знанням про вподобання людини-адресата, висувала припущення щодо подарунку, який би був бажаний і корисний [1]. Таку систему будемо називати рекомендаційною. Потенційні користувачі подібної системи – це будь-які користувачі мережі Internet, які мають складності з вибором подарунку якійсь іншій людині. Рекомендаційні системи є активною областю досліджень в галузі інтелектуального аналізу даних та машинного навчання. Ця тематика є ключовою для таких міжнародних наукових конференцій як RecSys (recsys.acm.org), SIGIR (sigir.org), KDD (kdd.org).

Методи розв'язку даної задачі відносяться до методів індуктивного навчання. Таких алгоритмів існує багато, тому важливим кроком розв'язання поставленої задачі є вибір алгоритму, який би якнайкраще відповідав запропонованій галузі досліджень.

В рамках цієї статті розглядаються алгоритми побудови дерева прийняття рішень та множини правил виведення. Необхідно зазначити, що з дерева прийняття рішень можна отримати правила виведення та навпаки.

Один з перших алгоритмів індуктивного навчання був розроблений Quinlan. Це алгоритм ID3 [2], а також його наступна модифікація J48 [3]. Окрім того слід відзначити методи Prism, запропонований Cendrowska [4], та PART, розглянутий Frank та Witten [5].

Важливим аспектом індуктивного навчання та функціонування рекомендаційних систем є використання нечіткої логіки для опрацювання неоднозначностей у вхідних та вихідних даних. Одним з важливих екземплярів цього класу алгоритмів є метод FuzzyID3, запропонований Umano та ін. [6]. Метод, який розглядається в цій роботі, ґрунтується якраз на модифікації алгоритму FuzzyID3.

Попередньо також був проведений аналіз існуючих рішень, тобто систем, які розв'язують подібну задачу. Були розглянуті наступні ресурси:

1. gifts.com.ua;
2. tutdar.com.ua;
3. market.yandex.ua/gifts.xml;
4. millionpodarkov.ru.

За результатами аналізу виявилось, що всі перелічені системи не орієнтуються на вибір подарунку під конкретну людину, а більше спрямовані на урахування приводу для подарунку. Окрім того, можна зауважити, що в цих системах використовуються експертні, жорстко визначені оцінки щодо того, яким категоріям користувачів варто рекомендувати конкретні подарунки. Даний факт не дозволяє говорити про гнучкість функціонування та ефективність надання рекомендацій існуючими системами.

2. Постановка задачі

Призначенням системи рекомендацій подарунків в Інтернет є надання рекомендації подарунків користувачам.

Нехай $A = \{(a_i, v_i, p_i)\}$ – множина входів, де a_i – це атрибут, v_i – значення атрибуту a_i, p_i – це фактор впевненості щодо того, що атрибут a_i має значення v_i. Для нашої предметної області атрибути a_i – це стать, вік, вартість подарунку, хобі, привід подарунків тощо. Значення фактору впевненості p_i лежить в проміжку від 0 до 1.

Множина виходів $B = (m_i, q_i)$, де m_i – це рекомендований товар/об'єкт, q_i – степінь впевненості в цьому рекомендованому товарі. Аналогічно до p_i, q_i також лежить в проміжку від 0 до 1.

В якості m_i в нашій предметній області виступають типи подарунків, наприклад, офіційний, косметика, побутова техніка, комп'ютерна техніка, квитки в театр тощо.

Необхідно розробити такий метод, який може побудувати відображення $f: A \to B$, яке якнайкраще відповідає індивідуальним очікуванням користувача.

3. Побудова дерева рекомендації

Оскільки кожен з алгоритмів [1-4] може надати тільки правила виведення чи дерево прийняття рішень, листками якого буде лише один варіант подарунку, втрачається ідея альтернативи вибору користувача. Наявність рекомендації лише одного подарунку позбавляє дарувальника різноманітності вибору, тобто якщо він вже такий подарунок зробив, то нічого іншого вибрати він не зможе. Саме тому у нагоді може стати метод, який будує дерево прийняття рішень, листки якого містять множину пар: «подарунок – фактор впевненості».

Одним з таких методів є алгоритм FuzzyID3 [5]. Він є модифікацією алгоритму ID3. Проте ID3 вибирає тестовий атрибут на основі приросту інформації, що обчислює ймовірність звичайних даних, однак FuzzyID3 обчислює ймовірність приналежності даних до деякої множини.

3.1 Нечіткі атрибути

Нехай маємо набір даних D, який складається з окремих кортежів. Кожен кортеж містить s значень для атрибутів $A_1, A_2, \ldots, A_s$, один клас з множини $C = \{C_1, C_2, \ldots, C_n\}$ та число μ, яке відповідає впевненості тому, що обраний клас C_i відповідає зазначеним атрибутам. Для нашої предметної області атрибути відповідають характеристикам адресата подарунку, а клас – це окрема категорія подарунків: C_1 – офіційний, C_2 – предмет гардероба, C_3 – гастрономічний, C_4 – для курця, C_5 – ювелірний виріб, C_6 – живий подарунок (тварина), C_7 – косметика, C_8 – побутова техніка, C_9 – комп'ютерна техніка, C_{10} – квитки в театр, кіно, виставку, C_{11} – сувенір, C_{12} – гроші.

Нехай є нечіткі множини $F_{i1}, F_{i2}, \ldots, F_{im}$ для кожного атрибуту A_i (значення m змінюється для кожного атрибуту).

Наприклад, в нашій предметної області маємо атрибут "вік" із можливими значеннями: «до18», «від 18 до 24», «від 25 до 34», «від 35 до 44», «від 45 до 54», «від 55 до 64», «старше 64». Коли користувач вказує певний вік адресату подарунку, то система повинна враховувати нечіткість цих даних. Таким чином ми створюємо нечіткі множини (fuzzy sets) для кожного можливого значення атрибуту «вік». Нижче наводяться приклади цих нечітких множин, де поруч з кожним елементом множини вказується його степінь приналежності цій множині.

- «до 18» = {1/до18, 0.3/18_24, 0.1/25_34};
- «від 18 до 24» = {0.4/до18, 1/18_24, 0.5/25_34, 0.1/35_44};
- «від 25 до 34» = {0.1/до18, 0.5/18_24, 1/25_34, 0.4/35_44};
- «від 35 до 44» = {0.1/18_24, 0.5/25_34, 1/35_44, 0.4/45_54};
- «від 45 до 54» = {0.1/25_34, 0.5/35_44, 1/45_54, 0.4/55_64};
- «від 55 до 64» = {0.1/35_44, 0.5/45_54, 1/55_64, 0.4/ старше64};
- «старше 64» = {0.2/45_54, 0.6/55_64, 1/старше64}

Нечіткі множини «близькі», «середні», «дальні» для атрибуту "стосунки" мають вигляд:

- «близькі» = {1/кохана людина, 0.9/член сім'ї, 0.8/близький друг, 0.4/родич, 0.2/ колега з роботи};
- «середні» = {0.4/кохана людина, 0.5/член сім'ї, 0.6/близький друг, 1/родич, 0.8/колега з роботі, 0.6/приятель, 0.3/інше};
- «дальні» = {0.7/родич, 0.9/колега з роботі, 1/приятель, 0.8/інше}.

Аналогічно будуються нечіткі множини значень й для інших атрибутів.

3.2 Карти атрибутів

Проте в описану вище схему важко вписати деякі атрибути. Наприклад, розглянемо атрибути, які послуговуються для визначення вподобань адресата подарунку. Ми дозволяємо користувачу обрати декілька вподобань (наприклад, спорт, йога, музика, відвідування театрів, допінг тощо), а також вказати наскільки це вподобання притаманне адресату подарунка (за шкалою від 1 до 5, де 1 – зовсім не притаманне, 5 – дуже притаманне).

Можна відзначити, що запропонована схема має два суттєвих недоліки. По-перше, кількість таких вподобань може бути дуже великою і це може вплинути на час побудови дерева прийняття рішень та надання подальших рекомендацій. По-друге, підхід, в якому кожне вподобання розглядається як окремий атрибут, не може бути ефективно масштабований (з точки зору додавання нових вподобань в систему пізніше).

Тому було запропоновано об'єднати усі вподобання в один атрибут, який ми назвали «хобі». Всі можливі вподобання ми розглянули з точки зору відповідності чотирьом критеріям: фізична чи розумова діяльність, пасивна чи активна діяльність. Причому ці критерії утворюють дві пари, які пов'язують між собою протилежні сутності (рис. 1).

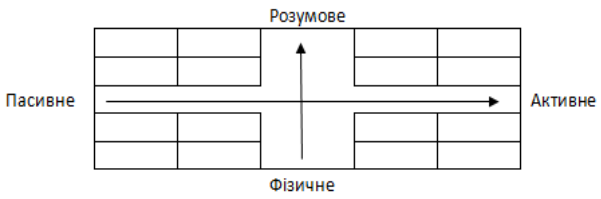

Рисунок 1 – Карта хобі

Тепер кожне вподобання адресата буде відображене на двовимірний простір, який визначається вимірами: «фізична-розумова діяльність» та «активна-пасивна діяльність». Наприклад, відвідування тренажерного залу бути тяжіти до фізичної та пасивної діяльності, а заняття екстремальними видами спорту – до фізичної та активної діяльності.

Для формалізації даного розподілення вподобань по критеріям були введені так звані «карти хобі» (рис. 1). Вони уявляють собою двовимірну матрицю розміром 4×4, де елементом матриці є значення від 0 до 1, що відповідає тому, наскільки сильно конкретне вподобання «присутнє» в певній точці двовимірного простору хобі.

Нарешті, всі вподобання адресата подарунку об'єднуються в одну матрицю шляхом підсумовування карт окремих хобі із врахуванням степеню притаманності (який ми позначимо через α_i; α_i набуває значень в проміжку від 0 до 1). Через H_i позначимо матрицю, яка відповідає карті i-го хобі. Тоді загальна карта хобі адресата вираховується за формулою (1).

$$H = \frac{1}{c}\sum_{i=1}^{n} \alpha_i \cdot H_i \qquad (1)$$

де $\alpha_i \in [0;5]$ – ранг i-ого хобі адресата, $i \in [1;n]$ (в нашій системі наразі $n = 19$ – загальна кількість вподобань), c – кількість вподобань, для яких $\alpha_i > 0$.

Аналогічно будується карта для атрибуту «привод подарунку» (рис. 2). Тут теж маємо дві пари протилежних критеріїв: приватний-публічний і традиційний-офіційний привід.

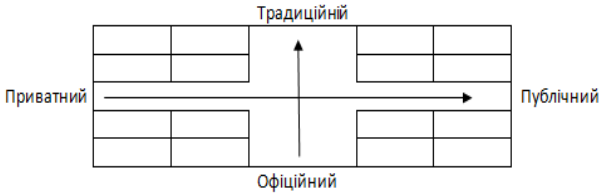

Рисунок 2 - Карта приводів

3.3 Алгоритм побудови дерева прийняття рішень

Алгоритм для побудови нечіткого дерева прийняття рішень для надання рекомендацій подарунків приймає на вхід множину навчальних даних D. Розглянемо кроки алгоритму.

Крок 1. Згенерувати корінь дерева як набір всіх даних D, задати ступінь впевненості μ для нечіткої множини, яка відповідає оцінці подарунка адресатом у навчальній вибірці. Обрати корінь як поточний вузол.

Крок 2. Якщо поточний вузол t з набором нечіткої множини даних D задовольняє одній з наступних умов:

- частка набору даних класу C_k більша або рівна θ_r:

$$\frac{\left|D^{C_k}\right|}{\left|D^{C}\right|} \geq \theta_r; \qquad (2)$$

- кількість наборів даних менше θ_n:

$$\left|D\right| < \theta_n; \qquad (3)$$

- немає більше атрибутів для класифікації,

то вважати цей вузол листком і розрахувати значення для ступенів впевненості для кожного класу з множини C (у (2) та (3) θ_r та θ_n є пара-

метрами керування, про які мова буде йти пізніше). Інакше

Крок 3.1. Для атрибутів A_1, A_2, …, A_s розрахувати прирости інформації $G(A_i, D)$ за формулою (4) та вибрати атрибут A_{max}, що максимізує їх.

$$G(A_i, D) = I(D) - E(A_i, D) \qquad (4)$$

де $I(D)$ – міра інформації, що розраховується за формулою (5), та $E(A_i, D)$ – очікувана інформація за формулою (7).

$$I(D) = -\sum_{k=1}^{n} p_k \cdot \log_2 p_k \qquad (5)$$

де p_k – фактор приналежності даних класу C_k в множині D:

$$p_k = \frac{\left|D^{C_k}\right|}{|D|} \qquad (6)$$

$$E(A_i, D) = \sum_{j=1}^{m} (p_{ij} \cdot I(D_{F_{ij}})) \qquad (7)$$

де p_{ij} – фактор приналежності значення j (нечіткого набору) серед усіх можливих значень атрибуту i:

$$p_{ij} = \frac{\left|D_{F_{ij}}\right|}{\sum_{j=1}^{m} \left|D_{F_{ij}}\right|} \qquad (8)$$

Крок 3.2. Розбити D на нечіткі множини D_1, D_2, …, D_m згідно з обраним атрибутом A_{max}, де значення ступеню впевненості буде розраховуватися як добуток ступеню приналежності множені D на значення $F_{max,j}$ для A_{max} в D. При цьому для всіх D_i атрибут A_{max} буде видалений з подальшого розгляду.

Деякі атрибути, а саме хобі та приводи подарунків використовують двовимірний масив в якості елементів нечітких множин (див. розділ 3.2). Це спричиняє деякі складності, зокрема, як перевести матрицю приналежності, наприклад, деякого хобі до окремого числа-значення впевненості μ_a (яке лежить в проміжку від 0 до 1). Було запропоновано використання зведення матриці (9).

$$\mu_{a_1} = \frac{1}{16} \sum_{i}^{n} \sum_{j}^{n} h_{ij} f_{ij}^{(a_1)} \in [0;1] \qquad (9)$$

де $H = \left\| h_{ij} \right\|$ – матриця хобі, $F_{a_1} = \left\| f_{ij}^{(a_1)} \right\|$ – матриця типізації хобі. Число 16 в знаменнику відповідає кількості комірок матриці хобі (нагадуємо, що вона має розмірність 4×4).

Крок 3.3. Згенерувати нові вузли t_1, t_2, ..., t_m для нечітких множин D_1, D_2, …, D_m та позначити нечіткою множиною $F_{max,j}$ ребро, що з'єднує вузли t_j та t.

Крок 3.4. Замінити D на D_j (j = 1, 2, …, m) та перейти на пункт 2.

3.3 Результати побудови дерева

Для побудови дерева можна використовувати різні значення параметрів θ_n та θ_r, відповідно до формул (2) та (3), які відповідають за деталізацію побудови дерева.

Приклад побудованого дерева зі значення $\theta_r = 0{,}7$ та $\theta_n = 100$ представлено на рисунку 3. Кожен листок дерева містить інформацію про тип подарунку та степінь приналежності вподобань опитуваних, що наведено у таблиці 1.

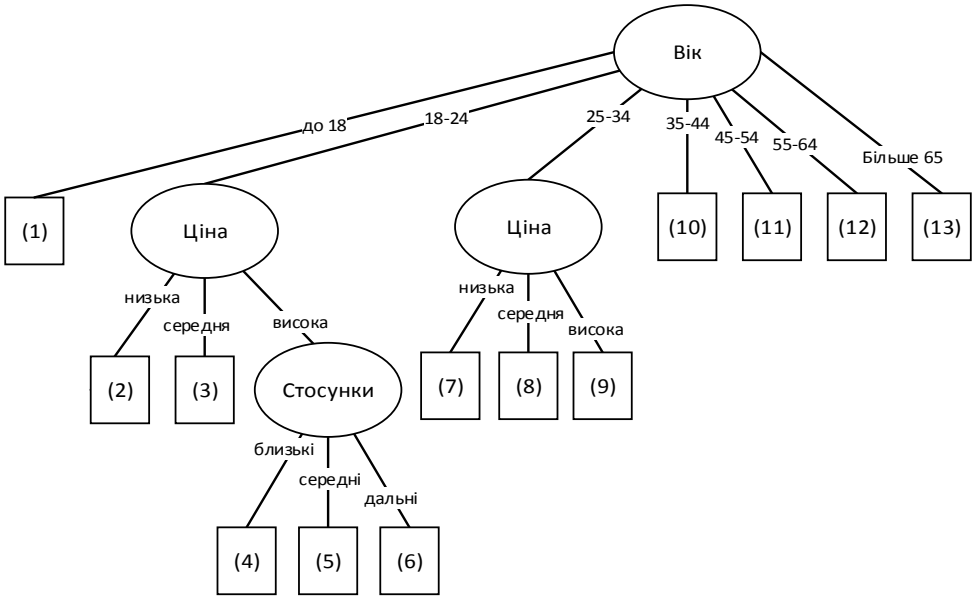

Рисунок 3 - Приклад нечіткого дерева прийняття рішень

Таблица 1 - Листки нечіткого дерева прийняття рішень

Клас / Листок	C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8	C_9	C_{10}	C_{11}	C_{12}
(1)	0,01	0,12	0,04	0,01	0,17	0,07	0,02	0,21	0,02	0,04	0,12	0,04
(2)	0,02	0,12	0,11	0,01	0,12	0,11	0,01	0,08	0,01	0,10	0,12	0,06
(3)	0,01	0,18	0,03	0,02	0,17	0,12	0,02	0,13	0,02	0,06	0,12	0,05
(4)	0,02	0,14	0,01	0,01	0,25	0,06	0,03	0,22	0,02	0,02	0,10	0,08
(5)	0,02	0,14	0,01	0	0,19	0,08	0,03	0,27	0,02	0,03	0,13	0,06
(6)	0,01	0,09	0	0	0,12	0,18	0,03	0,40	0	0	0,14	0
(7)	0,02	0,13	0,08	0,02	0,12	0,15	0,01	0,08	0,02	0,08	0,11	0,06
(8)	0,01	0,18	0,02	0,02	0,17	0,13	0,02	0,13	0,02	0,05	0,11	0,06
(9)	0,01	0,14	0	0,01	0,25	0,08	0,05	0,22	0,01	0,02	0,09	0,07
(10)	0,05	0,15	0,01	0,01	0,24	0,11	0,06	0,16	0,01	0,03	0,08	0,05
(11)	0,09	0,14	0	0,01	0,24	0,11	0,10	0,12	0,02	0,03	0,09	0,04
(12)	0,08	0,14	0	0,03	0,25	0,12	0,11	0,11	0,01	0,03	0,08	0,03
(13)	0,07	0,14	0	0,04	0,25	0,12	0,14	0,10	0,01	0,03	0,07	0,03

Були проведені експерименти зі значенням $\theta_r = 0{,}7$ та різними значеннями для $\theta_n = 100, 70, 50, 40, 30, 25, 20, 10$ (табл. 2).

Таблиця 2 - Результати експериментів

θ_n	Глибина дерева	Кількість вузлів	Максимальна кількість елементів в черзі
10	7	13530	8580
20	7	4356	2402
25	7	2565	1362
30	5	257	196
40	5	187	132
50	5	145	93
70	5	81	41
100	3	17	10

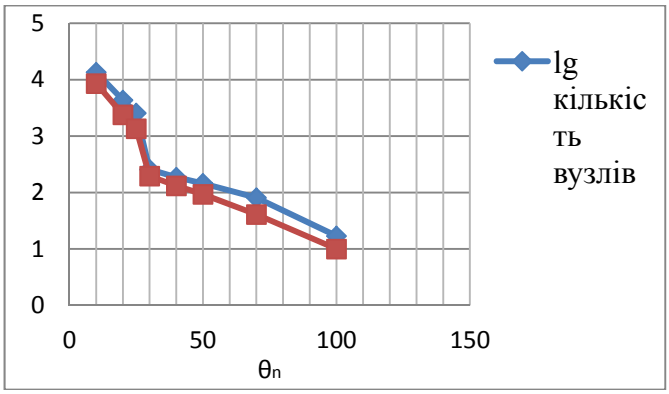

Рисунок 4 - Графіки залежності розмірності дерева від деталізації

Кількість вузлів дерева, як бачимо, зростає експоненціально зі зростанням деталізації, яке є обернено пропорційним до θ_n. З цього слідує, що чим детальнішим та більш точним має бути нечітке дерево прийняття рішень, тим більше часу необхідно витратити на його обрахунок.

4. Надання рекомендацій

Для надання рекомендацій на основі побудованого нечіткого дерева прийняття рішень, розглянемо наступний процес. На вхід подається один кортеж зі значеннями нечітких атрибутів (як описано в розділі 3.1). Для того, щоб отримати рекомендацію для нечіткого дерева прийняття рішень необхідно, починаючи з кореневого вузла перейти до тестового атрибуту у відповідному вузлі та повторити цю операцію допоки не дійшли до листового вузла. На відміну від звичайних дерев прийняття рішень, у нечітких деревах необхідно зробити переходи більше, ніж по одному ребру, враховуючи ступінь приналежності цього ребра.

Тепер потрібно вирішити три операції. По-перше, для об'єднання значень ступенів приналежності по ребрах, ми використовуємо добуток ступенів приналежності по ребрах одного шляху від кореня до листка. По-друге, отриманий ступінь на шляху помножується на фактор впевненості в листку для кожного з класів C_i. Нарешті, об'єднання ступенів приналежності кожного класу в нечіткому дереві прийняття рішень відбувається шляхом обрахунку суми ступенів для конкретного класу в усіх листках дерева. Додатково можна провести нормалізацію отриманих факторів приналежності для кожного класу.

Розглянемо приклад для нашої предметної області для побудованого дерева на рис. 3. Нехай користувач увів наступні значення атрибутів: стать = «жіноча», вік = «від 25 до 34», стосунки = «близький друг», вартість = «100 - 300» (інші атрибути не впливають на отримання рекомендації для цього дерева). На рис. 5 наведені ступені приналежності для кожного значення атрибутів (див. підписи до ребер дерева).

Для листка (2) дерева для класу C_1 отримуємо наступне значення фактору: $0{,}5\times0{,}4\times0{,}02 = 0{,}004$. Значення для всіх класів по всіх вузлах наведені в таблиці 3.

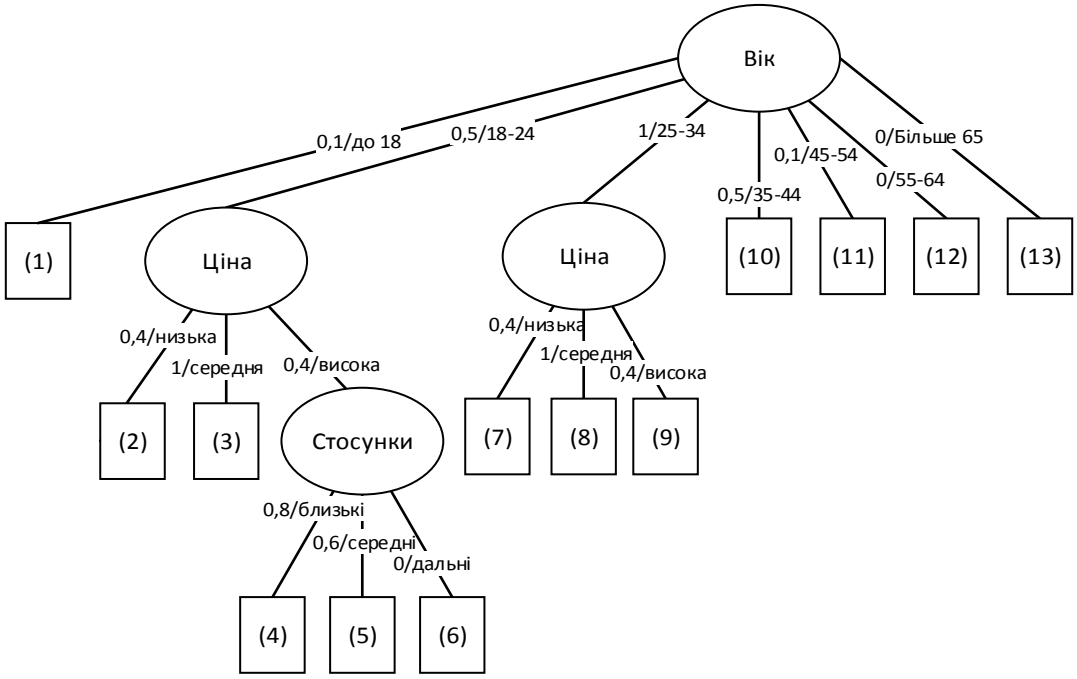

Рисунок 5 – Дерево прийняття рішень із факторами впевненості для вхідного кортежу

Таблиця 3 – Значення факторів впевненості по класах для дерева (рис. 5)

Клас / Листок	C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8	C_9	C_{10}	C_{11}	C_{12}
(1)	0,001	0,012	0,004	0,001	0,017	0,007	0,002	0,021	0,002	0,004	0,012	0,004
(2)	0,004	0,024	0,022	0,002	0,024	0,022	0,002	0,016	0,002	0,02	0,024	0,012
(3)	0,005	0,09	0,015	0,01	0,085	0,06	0,01	0,065	0,01	0,03	0,06	0,025
(4)	0,0032	0,0224	0,0016	0,0016	0,04	0,0096	0,0048	0,0352	0,0032	0,0032	0,016	0,0128
(5)	0,0024	0,0168	0,0012	0	0,0228	0,0096	0,0036	0,0324	0,0024	0,0036	0,0156	0,0072
(6)	0	0	0	0	0	0	0	0	0	0	0	0
(7)	0,008	0,052	0,032	0,008	0,048	0,06	0,004	0,032	0,008	0,032	0,044	0,024
(8)	0,01	0,18	0,02	0,02	0,17	0,13	0,02	0,13	0,02	0,05	0,11	0,06
(9)	0,004	0,056	0	0,004	0,1	0,032	0,02	0,088	0,004	0,008	0,036	0,028
(10)	0,025	0,075	0,005	0,005	0,12	0,055	0,03	0,08	0,005	0,015	0,04	0,025
(11)	0,009	0,014	0	0,001	0,024	0,011	0,01	0,012	0,002	0,003	0,009	0,004
(12)	0	0	0	0	0	0	0	0	0	0	0	0
(13)	0	0	0	0	0	0	0	0	0	0	0	0
Сума	0,0716	0,5422	0,1008	0,0526	0,6508	0,3962	0,1064	0,5116	0,0586	0,1688	0,3666	0,202
Нормалізовані суми	0,0222	0,1680	0,0312	0,0163	0,2016	0,1227	0,0330	0,1585	0,0182	0,0523	0,1136	0,0626

Отже, для вхідного кортежу даних найкращими рекомендаціями подарунків є ювелірний виріб (20%), предмети гардеробу (16,8%), побутова техніка (15,9%), тварина (12%) та сувеніри (11%). Окрім того, даний підхід дозволяє виявити найбільш небажані подарунки: подарунки для курця (1,6%), комп'ютерна техніка (1,8%), офіційний подарунок (2%).

5. Висновки

Була запропонована система, яка за певними критеріями та знанням про вподобання людини, висуває припущення щодо подарунку, який буде бажано отримати.

Зроблено аналіз наявних аналогів за результатами яких була обґрунтована необхідність створення такої системи, яка би спрощувала вибір подарунків користувачами тематичних веб сайтів (інтернет-магазинів).

Проведений аналіз відомих алгоритмів засвідчив необхідність розробки власного математичного методу. Було запропоновано модифікацію методу FuzzyID3 для урахування комплексних атрибутів (наприклад, вподобання та привід подарунку). Запропонований метод дозволяє легке масштабування - врахування додаткових атрибутів.

Поза межами уваги лишились два важливих питання. По-перше, запропонований метод має суттєвий недолік, який полягає у підсиленні типових рекомендацій. Для нашої предметної області це означає неможливість рекомендування оригінальних подарунків. По-друге, при практичному застосуванні системи ми стикнемось із великою кількістю категорій подарунків, які варто було би об'єднати у деяку таксономію. Тож необхідно модифікувати запропонований алгоритм для можливості роботи із класами категорій подарунків.

Список літератури

1. Знахуренко В.П. Система надання рекомендацій щодо вибору подарунків. Тези IV Всеукраїнської заочної науково-практичної конференції «Сучасні інформаційні технології» Київ. 2013. – 115 с.
2. R. Quinlan (1986). Induction of decision trees. Machine Learning. 1(1):81-106.
3. Ross Quinlan (1993). C4.5: Programs for Machine Learning, Morgan Kaufmann Publishers, San Mateo, CA.
4. J. Cendrowska (1987). PRISM: An algorithm for inducing modular rules. International Journal of Man-Machine Studies. 27(4):349-370.
5. Eibe Frank, Ian H. Witten: Generating Accurate Rule Sets Without Global Optimization. In: Fifteenth International Conference on Machine Learning, 144-151, 1998.
6. Umano M., Okamoto H., Hatono I., Tamura H., Kawachi F., Umedzu S., Kinoshita J. Fuzzy decision trees by fuzzy ID3 algorithm and its application to diagnosis systems. - Fuzzy Systems, 1994. IEEE World Congress on Computational Intelligence., Proceedings of the Third IEEE Conference on, 2113 - 2118 vol.3