

**СИСТЕМИ З НЕЧІТКОЮ ЛОГІКОЮ В ЗАДАЧАХ
ЕКСПЕРТНОГО ОЦІНЮВАННЯ**

Л.А. БОЙКО, Ю.П. ЗАЙЧЕНКО

Проведено експериментальні дослідження системи з нечіткою логікою — NEFCLASS — для класифікації запропонованих на конкурс інноваційних проєктів з державного фінансування, а також для ранжування їх за необхідністю фінансування.

ВСТУП

Експертне оцінювання в останні роки набуло широкого розвитку та застосування в задачах оцінки проєктів, поданих на конкурси, і в задачах прийняття стратегічних рішень у різних сферах економіки та фінансів тощо.

Рішення, які приймаються експертами, характеризуються наявністю неповної, нечіткої та суперечливої інформації, використанням, крім кількісних, ще й якісних критеріїв. При цьому знання експертів носять слабо структурований характер.

Розв'язання багатьох проблем при експертному оцінюванні тісно пов'язане з обробкою неповної, неточної і суперечливої інформації (наприклад, у задачах аналізу фінансових ринків, вибору стратегії розвитку підприємства тощо). Для вирішення таких проблем необхідно застосовувати неструктуровані процедури вироблення рішень, які використовують знання та інтуїцію людини-експерта.

Для моделювання й обробки нечіткої лінгвістичної та якісної інформації в останні роки використовується апарат нечітких множин та відношень, система нечіткого логічного висновку та нечіткі нейронні мережі.

Метою даної статті є дослідження систем з нечіткою логікою в задачах експертного оцінювання, зокрема при оцінюванні проєктів, поданих на конкурс.

ПОСТАНОВКА ТА ФОРМАЛІЗАЦІЯ ЗАДАЧІ

Розглянемо задачу класифікації інноваційних проєктів за якісними показниками (критеріями) із застосуванням нечіткої нейромережі.

Предметна область — це експертні оцінки науково-дослідних робіт, що подаються на конкурс з метою отримання держбюджетного фінансування. Оцінювання проектів здійснюється за такими показниками: новизна та актуальність, наукова цінність, соціально-економічна значимість, загальний рівень розробки.

Задача полягає у визначенні, до якого класу віднести науково-дослідну роботу: фінансувати її, частково фінансувати чи взагалі не фінансувати.

Існує досить значний клас змінних та понять, які не можуть бути охарактеризовані в рамках класичної теорії оцінювання. До них відносяться змінні, що характеризують розмір (дуже маленький, маленький, середній, великий, дуже великий), рівень (низький, середній, високий), а також якісні властивості системи. Принциповою властивістю цих змінних є існування розмитої границі між різними градаціями тієї чи іншої якості. Так, доволі складно точно визначити пограничний зріст, що відрізняє високу людину від людини середнього зросту. В реальності границя між цими якісними поняттями має нечіткий характер. Для описання таких понять у нечіткій математиці використовуються нечіткі множини, характеристичні функції яких можуть приймати значення з усього інтервалу від 0 до 1, тобто кожна точка характеризується ступенем її належності до нечіткої множини. Такий підхід дозволяє більш гнучко реалізовувати метод експертних оцінок, ніж традиційна теорія ймовірностей. Експерту необхідно формалізувати свої оцінки можливих значень величини в термінах задання характеристичної функції множини значень, які вона може приймати.

Розглянемо формалізовану постановку даної задачі класифікації.

Вхідні дані. Маємо n інноваційних проектів, де $n = \overline{1, N_0}$. Кожний проект характеризується показниками якості Q_j , $j \in \overline{1, J}$, а кожний показник Q_j є лінгвістичною змінною. Позначимо $\{Q_{js}\}$ кількість рівнів градацій j -го показника $s \in \overline{1, S}$. Рівні градації описуються, наприклад, такими лінгвістичними значеннями:

$$s_1 = \text{very small (VS)}; s_2 = \text{small (S)}; s_3 = \text{middle (M)};$$

$$s_4 = \text{high (H)}; s_5 = \text{very high (VH)}.$$

Лінгвістичні значення Q_j у загальному випадку можуть бути довільними. Для визначеності виберемо їх трикутними або трапецеїдальними (див. рисунок).

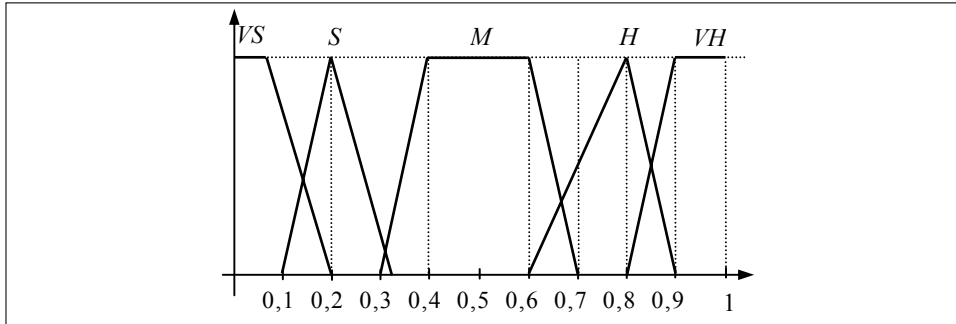
Експертна інформація по кожному експерту k задається у вигляді таких правил класифікації:

$$R_1: \text{якщо } Q_1 \in A_{11}^k, Q_2 \in A_{21}^k, \dots, Q_j \in A_{j1}^k, \text{ то } u \text{ належить до класу } \Omega_1;$$

$$R_i: \text{якщо } Q_1 \in A_{1i}^k, Q_2 \in A_{2i}^k, \dots, Q_j \in A_{ji}^k, \text{ то } u \text{ належить до класу } \Omega_i;$$

$$R_m: \text{якщо } Q_1 \in A_{1m}^k, Q_2 \in A_{2m}^k, \dots, Q_j \in A_{jm}^k, \text{ то } u \text{ належить до класу } \Omega_m,$$

де $A_{11}^k, \dots, A_{j1}^k, \dots, A_{ji}^k, A_{jm}^k$ — лінгвістичні значення показників Q_1, \dots, Q_j (наприклад, S, M, H).



Вид функції залежності

Вихідні дані. Нехай множина класів оцінювання проектів за ступенем досягнення показників цільових функцій така:

- 1) Q_1 — песимістична оцінка (не фінансувати проект);
- 2) Q_2 — раціональна оцінка (фінансувати частково);
- 3) Q_3 — оптимістична оцінка (фінансувати повністю).

Кожен клас Ω_i описується своєю функцією належності $\mu(\Omega)$.

$\Omega_1, \dots, \Omega_i, \dots, \Omega_m$, де $i = \overline{1, 3}$ — множина класів оцінок цільових функцій відповідного проекту k -го експерту.

Мета роботи системи. На основі наявної нечіткої інформації за класифікацією базових проектів k -м експертом налагодити базу правил і виконати навчання функцій належності (ФН) для лінгвістичних змінних A_{ij}^k .

У цій задачі доцільно використати нечіткий перцептрон типу NEFClass, що використовує логічний висновок Мамдані [4, 5]. Експертна нечітка система будується за частковими знаннями про зразки, тобто за особистою інформацією експертів про класифікацію попередніх проектів.

АЛГОРИТМ НАСТРОЮВАННЯ ПАРАМЕТРІВ НЕЙРОННОЇ СИСТЕМИ

Налагодження нейронної мережі проходить у два етапи.

Параметри, які використовуються для настроювання:

S — кількість лінгвістичних значень для кожного показника Q_j ; k_{\max} — максимальне число вузлів правил, які можуть бути створені у прихованому шарі; σ — величина швидкості навчання; t — бажаний результат; E — помилка системи; $\varepsilon_{\text{доп}}$ — задана помилка системи; n — ітерація у процесі навчання.

Користувач (проектувальник) повинен визначити число лінгвістичних значень S для кожного показника Q_j і задати k_{\max} — максимальне число вузлів правил, які можуть бути створені у прихованому шарі.

Нехай є набір навчальних зразків $L = \{(p_1, t_1), \dots, (p_i, t_i), \dots, (p_l, t_l)\}$, кожний з яких складається з набору вхідних показників l -го проекту $\{Q_1^{(l)}, \dots, Q_j^{(l)}, \dots, Q_J^{(l)}\}$ і бажаної класифікації $t_l \in \Omega_i$, тобто $p_l \in R^J$.

Перший етап. Алгоритм навчання бази правил складається з однотипних ітерацій, на кожній з яких вводиться правило виводу.

1. Обираємо наступний зразок $(p_l, t_l) \in L$.
2. Для кожного вхідного нейрона $Q_j \in U_1$ (вхідний шар). Знаходимо таку функцію належності $\mu_{j1}^{(p)}$, для якої $\mu_{j1}(Q_j^{(l)}) = \max_{v=1,s} \{\mu_{jv}(Q_j^{(l)})\}$.
3. Якщо число вузлів правил $R < k_{\max}$, і не існує такого вузла правила R , для якого $w(Q_1^{(l)}, R) = \mu_{11}$; $w(Q_j^{(l)}, R) = \mu_{ji}$; $w(Q_J^{(l)}, R) = \mu_{Js}$, то створюємо новий вузол R_i і з'єднуємо його з вихідним вузлом Ω_m . Якщо $t_l \in \Omega_m$, приймаємо вагу зв'язку $w(R_i, \Omega_m) = 1$.
4. Якщо ще залишилися необроблені зразки в L і $k < k_{\max}$, то переходимо на крок 1 наступної ітерації. В іншому випадку — на крок 5.
5. Формуємо базу правил за однією з трьох процедур:
 - «Просте» навчання правил: залишаємо тільки перші $k = k_{\max}$ правил.

• «Краще» навчання правил: обробляємо зразки з L і накопичуємо активації (кількість активацій) кожного нейрона правил. Якщо нейрон R показує більшу кількість накоплених активацій для класу Ω_j , ніж для класу Ω_i , який був визначений для наслідку правила (початковий), то змінюємо наслідок R на Ω_j , тобто з'єднуємо нейрон R з нейроном виходу Ω_j .

Продовжуємо далі обробку зразків в L та для кожного нейрона правил обчислюємо показник ефективності

$$V_R = \sum_{p \in L} a_R^{(p)} l_p,$$

де $l_p = \begin{cases} 1, & \text{якщо зразок (проект } p) \text{ класифікований вірно,} \\ 0 & \text{в іншому випадку.} \end{cases}$

Залишаємо k нейронів правил з найвищим значенням V_R та видаляємо інші з бази правил.

- «Краще для кожного класу» навчання правил: діє так само як і «краще», але для кожного класу Ω_j залишаємо кращі $r = \left\lceil \frac{k}{m} \right\rceil$ $m = 3$ правил, наслідки яких складають клас Ω_j .

Другий етап. Навчання параметрів функцій належності нечітких множин — налаштувати параметри функцій належності нечітких множин, які описують відповідні лінгвістичні значення вхідних змінних μ_{jv} . Кожна функція належності характеризується такими параметрами (a_j, c_j, b_j) :

1. Обираємо наступний зразок (проект) (p, t) , подаємо його на вхід системи і визначаємо вихідний вектор $\Omega = \{\Omega_1, \dots, \Omega_i, \dots, \Omega_m\}$.
2. Для кожного вихідного нейрона Ω_i визначаємо значення помилки класу $\delta_{\Omega_i} = t_i - a_{\Omega_i}$, де Ω_i — сигнал вихідного нейрона.
3. Для кожного нейрона правил R , у якого a_R (вихід) > 0 :
 - а) визначаємо значення δ_R правил $\delta_R = a_R(1 - a_R) \sum_{\Omega \in U_3} w(R, \Omega) \delta_{\Omega}$;
 - б) знаходимо таке Q'_j , для якого $w(Q'_j, R)(Q_j) = \min_{Q_i \in U_1} \{w(Q_i, R)(Q_i)\}$;
 - в) для нечіткої множини $W(Q'_j, R)$ визначаємо δ_{a_j} , δ_{b_j} і δ_{c_j} , використовуючи параметр швидкості навчання σ :

$$\delta_{b_j} = \sigma \delta_R (c_j - a_j) \text{sign}(Q'_j - b_j),$$

$$\delta_{a_j} = -\sigma \delta_R (c_j - a_j) + \delta_{b_j},$$

$$\delta_{c_j} = \sigma \delta_R (c_j - a_j) + \delta_{b_j}.$$

Застосовуємо ці зміни для обчислення нових значень a_j, c_j, b_j .

4. Обчислюємо помилку правила: $E = a_R(1 - a_R) \sum_{\Omega_i \in U_3} (2w(R, \Omega_i) - 1) \delta_{\Omega_i}$.

Кінець k -ї ітерації.

Зазначені ітерації навчання повторюємо доти, поки не буде виконуватись критерій зупинки:

1. Помилка E не зменшується протягом n ітерацій.
2. Помилка досягла деякого заданого значення $\varepsilon_{\text{доп}}$.

АЛГОРИТМ ЕКСПЕРТНОЇ СИСТЕМИ

Аналогічним чином налаштовується база правил і навчаються параметри функцій належності для всіх інших експертів. Опишемо роботу алгоритму із використанням логічного висновку Мамдані.

Введемо параметр ν_k — компетентність k -го експерта.

1. Ініціалізуємо значення всіх параметрів. Подаємо на вхід системи для k -го експерта новий проект, який описується набором $\{Q_1^H, \dots, Q_j^H, \dots, Q_J^H\}$.
2. Завантажуємо правила.
3. Для кожного правила реалізуємо такі дії:
 - 3.1. Розраховуємо реакції всіх вхідних нейронів $\mu_{js}(Q_j^H)$, $s = \overline{1, S}$, $j \in J$, тобто виконується процедура фазифікації.
 - 3.2. Обчислюємо вихід кожного нейрона правил R (рівень значущості правила)

$$a_R(Q^H) = \min_{\substack{j \in J \\ Q_j \in U_1}} \{w(Q_j, R)(Q_j)\}.$$

3.3. Обчислюємо класифікацію зразка $\{Q_j^H\}$

$$Q^H \in \Omega_i, \text{ якщо } W^k(R, \Omega_i) = \max_{\Omega} W(R, \Omega)(a_R).$$

3.4. Повторюємо дії пунктів 3.1...3.3 k разів.

4. Виконуємо дефазифікацію отриманих нечітких виходів по всім K експертам з урахуванням їх компетентності (узагальнена групова класифікація)

$$\{Q_j^H\} \in \Omega_i, \text{ якщо } \sum_{k=1}^K W^k(R, \Omega_i)v^k \rightarrow \max_{\Omega} \sum_{k=1}^K W^k(R, \Omega)v^k.$$

5. На виході отримуємо чітке значення вихідного показника.

РЕЗУЛЬТАТИ РОБОТИ СИСТЕМИ НЕЧІТКОЇ КЛАСИФІКАЦІЇ

У даній роботі поставлена задача на основі відомої нечіткої експертної інформації за класифікацією нових проектів k -м експертом провести навчання бази правил та функцій належності лінгвістичних змінних.

Розроблено та відлагоджено програму, яка реалізує запропонований алгоритм навчання системи та наступної класифікації проектів.

Покажемо роботу створеного програмного продукту та ефективність даної системи на конкретному прикладі.

Вхідні дані, за якими буде навчатись система NefClass, це вибірка із 120 зразків (табл. 1). Вони представляють собою експертні оцінки науково-дослідних робіт НТУУ «КПІ», поданих на конкурс з фінансування розробок за рахунок державного бюджету в 1999 р. Задача полягає у визначенні, до якого класу віднести науково-дослідну роботу: фінансувати її, частково фінансувати чи взагалі не фінансувати. Як вхідні параметри беремо критерії

- X1 — новизна та актуальність роботи;
- X2 — наукова цінність;
- X3 — соціально-економічна значимість;
- X4 — загальний рівень розробки.

Всі вказані вхідні параметри вимірюються якісно.

Як вихід зразка беремо розбиття на три класи у порядку збільшення фінансування проектів.

- Перший клас (100) — «не фінансувати проект»;
- Другий клас (010) — «частково фінансувати проект»;
- Третій клас (001) — «повністю фінансувати проект».

Створення бази правил. Як нечітке розбиття вхідних параметрів на основі досвіду експерта та ступеня важливості параметрів візьмемо таке: X1, X2, X3, X4 складаються з трьох трикутних нечітких множин, які описуються

лінгвістичними термами «погано», «добре», «гарно». Для побудови бази правил потрібно розглянути коефіцієнт кореляції. Показником взаємозалежності між величиною $X = \{X_1, X_2, X_3, X_4\}$ і вихідною величиною Y можна вибрати коефіцієнт парної кореляції.

$$k = \frac{\sum_{i=1}^N (X_{ji} - \bar{X}_{ji})(Y_i - \bar{Y}_i)}{\sqrt{\sum_{i=1}^N (X_{ji} - \bar{X}_{ji})^2 \sum_{i=1}^N (Y_i - \bar{Y}_i)^2}},$$

де $\bar{X}_{ji} = \frac{1}{N} \sum_{i=1}^N X_{ji}$; $\bar{Y}_i = \frac{1}{N} \sum_{i=1}^N Y_i$; N — число реалізацій.

Таблиця 1. Навчальна вибірка

X1	X2	X3	X4	Клас
3,1	3,5	3,4	2	100
2,9	3,0	3,4	2	100
1,7	3,2	3,3	2	100
2,6	3,1	3,5	2	100
...				
7,0	5,2	4,7	4	010
6,4	5,2	4,5	5	010
6,9	5,1	4,9	5	010
5,5	4,3	4,0	3	010
...				
6,3	7,3	6,0	5	001
8,8	6,7	7,1	8	001
7,1	7,0	5,9	5	001
8,3	6,9	5,6	8	001
...				
7,7	7,0	6,1	7	001
6,3	6,4	5,6	5	001
6,4	7,1	5,5	8	001
8,0	7,0	5,8	8	001

Після аналізу вхідних даних за допомогою експертів було складено базу правил, наведену в табл. 2.

Обґрунтування параметрів програми. З урахуванням попередніх досліджень обрані такі оптимальні параметри алгоритму навчання.

Параметри логічного висновку правил:

- Максимальна кількість правил — $3^4 = 81$.
- Кількість вхідних змінних — 4.
- Вид правил (наведено у табл. 2).
- Метод логічного висновку — Мамдані.
- Кількість лінгвістичних термів — 3.

Таблиця 2. База правил

1	IF	X ₁	is	small	and	X ₂	is	small	and	X ₃	is	small	and	X ₄	is	small	THEN	Class_1
2	IF	X ₁	is	small	and	X ₂	is	small	and	X ₃	is	medium	and	X ₄	is	medium	THEN	Class_1
3	IF	X ₁	is	medium	and	X ₂	is	small	and	X ₃	is	small	and	X ₄	is	small	THEN	Class_2
4	IF	X ₁	is	small	and	X ₂	is	medium	and	X ₃	is	small	and	X ₄	is	medium	THEN	Class_1
5	IF	X ₁	is	small	and	X ₂	is	small	and	X ₃	is	small	and	X ₄	is	medium	THEN	Class_1
6	IF	X ₁	is	small	and	X ₂	is	small	and	X ₃	is	medium	and	X ₄	is	small	THEN	Class_1
7	IF	X ₁	is	large	and	X ₂	is	medium	and	X ₃	is	medium	and	X ₄	is	medium	THEN	Class_2
8	IF	X ₁	is	medium	and	X ₂	is	medium	and	X ₃	is	medium	and	X ₄	is	medium	THEN	Class_2
9	IF	X ₁	is	medium	and	X ₂	is	small	and	X ₃	is	medium	and	X ₄	is	large	THEN	Class_2
10	IF	X ₁	is	medium	and	X ₂	is	small	and	X ₃	is	small	and	X ₄	is	small	THEN	Class_2
11	IF	X ₁	is	medium	and	X ₂	is	medium	and	X ₃	is	small	and	X ₄	is	small	THEN	Class_2
12	IF	X ₁	is	medium	and	X ₂	is	medium	and	X ₃	is	medium	and	X ₄	is	small	THEN	Class_2
13	IF	X ₁	is	medium	and	X ₂	is	medium	and	X ₃	is	medium	and	X ₄	is	large	THEN	Class_2
14	IF	X ₁	is	medium	and	X ₂	is	large	and	X ₃	is	medium	and	X ₄	is	small	THEN	Class_2
15	IF	X ₁	is	medium	and	X ₂	is	medium	and	X ₃	is	medium	and	X ₄	is	small	THEN	Class_2
16	IF	X ₁	is	large	and	X ₂	is	medium	and	X ₃	is	large	and	X ₄	is	large	THEN	Class_3
17	IF	X ₁	is	medium	and	X ₂	is	large	and	X ₃	is	large	and	X ₄	is	medium	THEN	Class_3
18	IF	X ₁	is	large	and	X ₂	is	large	and	X ₃	is	large	and	X ₄	is	large	THEN	Class_3
19	IF	X ₁	is	large	and	X ₂	is	large	and	X ₃	is	large	and	X ₄	is	medium	THEN	Class_3
20	IF	X ₁	is	large	and	X ₂	is	medium	and	X ₃	is	large	and	X ₄	is	large	THEN	Class_3
21	IF	X ₁	is	medium	and	X ₂	is	large	and	X ₃	is	medium	and	X ₄	is	large	THEN	Class_3
22	IF	X ₁	is	large	and	X ₂	is	large	and	X ₃	is	medium	and	X ₄	is	medium	THEN	Class_3
23	IF	X ₁	is	medium	and	X ₂	is	large	and	X ₃	is	large	and	X ₄	is	large	THEN	Class_3
24	IF	X ₁	is	large	and	X ₂	is	medium	and	X ₃	is	large	and	X ₄	is	medium	THEN	Class_3
25	IF	X ₁	is	medium	and	X ₂	is	medium	and	X ₃	is	large	and	X ₄	is	medium	THEN	Class_2

- Вид функцій належності — трикутні.

Параметри навчання та класифікації:

- Кількість ітерацій — 100.
- Розмірність навчальної вибірки — 120 зразків.
- Кількість некласифікованих даних — 5 експертів по 20 проектів.
- Процедура навчання бази правил — «краще для кожного класу».
- Швидкість навчання функцій належності — $\sigma = 0,01$.
- Функція активації для виходу — «максимум».

Параметри класифікації з урахуванням компетентності експертів:

- Кількість експертів — 5 чоловік.
- Кількість проектів — 20.
- Компетентність експертів
1) 0,5; 2) 0,7; 3) 0,9; 4) 0,4; 5) 0,6.

Експеримент. (Функція активації виходу — максимум. Процедура для навчання бази правил — «краще»). Після проведення процедури навчання було вірно розпізнано 114 із 120 зразків. Відповідно показник мережі складає 95,00 % при середньому значенні похибки 0,109585. Це дуже добрий показник для мережі.

База правил скоротилась з 25 до 12 за активаціями виходів правил. У процесі налаштування бази змінились виходи таких правил:

- Вихід правила 3 змінився з 1 на 2.
- 16 з 2 на 3.
- 25 з 3 на 2.

У процесі навчання функцій належності змінились їх параметри, кінці та середина трикутних функцій належності були зміщені, функції розширились.

Тепер проведемо класифікацію вхідних некласифікованих даних, а саме експертних оцінок 20-ти запропонованим на конкурс з державного фінансування проектів, які виставлені п'ятьма експертами в галузі оцінювання науково-дослідних робіт.

Після проведення класифікації цих даних ми отримали результати (табл. 3), які потім класифікувалися з урахуванням компетентності кожного експерта. Всі результати класифікації та їх порівняння з реальними наведено у табл. 4 та 5.

У результаті за даними навчання система отримала значення для класифікації, виходячи з раніше налаштованої бази правил. Ми бачимо, що деякі експерти віднесли один і той же проект до різних класів. Враховуючи компетентність експертів, визначаємо, до якого класу відноситься проект.

З цього експерименту можна зробити висновок, що система NEFCLASS є достатньо ефективною, навіть при неповноті та суперечливості інформації. А розроблена нами процедура врахування компетентності експертів підвищує ефективність класифікації в експертній системі NEFCLASS.

Таблиця 3. Результати класифікації за допомогою NefClass

Номер проекту	Клас	Експерти					Кінцева класифікація
		1	2	3	4	5	
		Вага					
1	1	0,745	0,687	0,630	0,687	0,687	Клас 1
	2	0	0	0	0	0	
	3	0	0	0	0	0	
2	1	0,253	0,431	1	0,333	1	1
	2	0,066	0,066	0	0	0	
	3	0	0	0	0	0	
3	1	1	0,959	1	0,826	0,692	1
	2	0	0	0	0	0,176	
	3	0	0	0	0	0	
4	1	0,687	0,737	0,585	0,514	0,781	1
	2	0	0	0,132	0,132	0	
	3	0	0	0	0	0	
5	1	0,870	0,737	0,901	0,959	0,696	1
	2	0	0	0	0	0	
	3	0	0	0	0	0	
6	1	0,901	0,737	0,959	1	1	1
	2	0	0	0	0	0	
	3	0	0	0	0	0	
7	1	0,648	0,901	0,603	1	0	1
	2	0	0	0,043	0	0,765	
	3	0	0	0	0	0	
8	1	0,031	0,901	0	0,060	0	1
	2	0,455	0	0,626	0,433	0,239	
	3	0	0	0,131	0	0	
9	1	0	0,297	0,164	0	0,248	1
	2	0,282	0,066	0,696	0,368	0,630	
	3	0	0	0	0	0	
10	1	0	0	0	0,585	0	1
	2	0,765	0,752	0,573	0	0,647	
	3	0,187	0	0	0	0,020	
11	1	0	0	0	0	0	2
	2	0,312	0,640	0,696	0,697	0,169	
	3	0,216	0,174	0	0	0,243	
12	1	0	0	0	0,031	0	2
	2	0,697	0,697	0,670	0,585	0,605	
	3	0,076	0,298	0	0	0,020	
13	1	0	0	0	0	0,031	2
	2	0,147	0	0,630	0,697	0,640	
	3	0,478	0,354	0	0,076	0	

Закінчення табл. 3

14	1	0	0	0	0	0	2
	2	0	0,696	0	0,477	0,202	
	3	0,697	0	0,605	0	0,697	
15	1	0	0	0	0	0	2
	2	0	0,257	0	0,689	0	
	3	0,521	0,697	0,604	0,217	0,691	
16	1	0	0	0	0	0	3
	2	0,093	0,093	0	0	0	
	3	0,765	0,697	0,521	0,688	0,689	
17	1	0	0	0	0	0	3
	2	0	0,056	0	0	0	
	3	0,521	0,410	0,626	0,604	0,697	
18	1	0	0	0	0	0	3
	2	0,169	0	0	0	0	
	3	0,744	0,697	0,605	0,911	0,648	
19	1	0	0	0	0	0	3
	2	0	0,267	0	0	0	
	3	0,821	0,697	0,648	0,562	0,735	
20	1	0	0	0	0	0	3
	2	0	0	0	0	0	
	3	1	0,800	0,648	1	0,648	

Таблиця 4. Результати класифікації з урахуванням компетентності експертів

Номер проекту	Результати NefClass	Експерти (компетентність, v_k)					Сумарна оцінка	Кінцева класифікація
		1 (0,5)	2 (0,7)	3 (0,9)	4 (0,4)	5 (0,6)		
	Клас	Оцінка з урахуванням компетентності експерта					Клас	
1	1	0,3725	0,4809	0,567	0,2748	0,4122	2,1074	1
2	1	0,1265	0,3017	0,9	0,1332	0,6	2,0614	1
3	1	0,5	0,6713	0,9	0,3304	0,4152	2,8169	1
4	1	0,3435	0,5159	0,5265	0,2056	0,4686	2,0601	1
5	1	0,435	0,5159	0,8109	0,3836	0,4176	2,563	1
6	1	0,4505	0,5159	0,8631	0,4	0,6	2,8295	1
7	1	0,324	0,6307	0,5427	0,4		1,8974	1
	2			0,0387		0,459	0,4977	
8	1	0,0155	0,6307		0,024		0,6702	2
	2	0,2275		0,5634	0,1772	0,1434	1,1115	
9	1		0,2079	0,1476		0,1488	0,5043	2
	2	0,141	0,0462	0,6264	0,1472	0,378	1,3388	

Закінчення табл. 4

10	1				0,234		0,234	
	2	0,3825	0,5264	0,5157		0,3882	1,8128	2
11	2	0,3485	0,448	0,6264	0,2788	0,1014	1,8031	2
	3	0,038	0,1218			0,1458	0,3056	
12	2	0,156	0,4879	0,603	0,234	0,363	1,8439	2
13	2	0,3485		0,567	0,2788	0,384	1,5783	2
	3	0,038	0,2478		0,0304		0,3162	
14	2	0,0735	0,4872		0,1908	0,1212	0,8727	
	3	0,239		0,5445		0,4182	1,2017	3
15	2		0,1799		0,2756		0,4555	
	3	0,3485	0,4879	0,5436	0,0868	0,4146	1,8814	3
16	3	0,3825	0,4879	0,4689	0,2752	0,4134	2,0279	3
17	3	0,2605	0,287	0,5634	0,2416	0,4182	1,7707	3
18	3	0,372	0,4879	0,5445	0,3644	0,3888	2,1576	3
19	3	0,4105	0,4879	0,5832	0,2248	0,441	2,1474	3
20	3	0,5	0,7	0,5832	0,4	0,3888	2,572	3

Для більш детального аналізу отриманих даних наведемо порівняльну таблицю результатів різних експериментів та реальних значень класифікації даних проектів, наданих відділом експертного оцінювання науково-дослідних робіт НТУУ «КПІ» (табл. 5).

Таблиця 5. Порівняння результатів класифікацій

Клас	Класифікація			
	Експеримент 1		Реальні значення	
Клас 1 «Не фінансувати»	Проект 7	1,8974	Проект 3	18
	4	2,0601	4	19
	2	2,0614	5	19
	1	2,1074	6	19
	5	2,5630	1	22
	3	2,8169	7	24
	6	2,8295	2	26
Клас 2 «Частково фінансувати»	8	1,1115	9	30
	9	1,3388	8	33
	13	1,5783	11	38
	11	1,8031	14	38
	10	1,8128	10	40
Клас 3 «Фінансувати повністю»	12	1,8439	12	40
	14	1,2017	13	40
	17	1,7707	16	50
	15	1,8814	17	50
	16	2,0279	19	50
	19	2,1474	15	51
	18	2,1576	20	51
20	2,5720	18	53	

З таблиці порівнянь результатів класифікацій видно, що у 3-му класі результати роботи системи NEFCLASS практично співпадають з результатами експертів.

Порівнявши отримані дані за класифікацією NEFCLASS та звичайної рейтингової системи, можна зробити висновок, що завдяки навчання бази правил та настроюванню параметрів функцій належності система NEFCLASS дає більш точний результат класифікації проектів. Продемонструємо це на проектах № 18, 19 та 20 за вхідними даними трьох експертів (табл. 6).

Таблиця 6. Вхідні дані найкращих проектів

NEFCLASS												
Експерт	Проект 18				Проект 19				Проект 20			
	X1	X2	X3	X4	X1	X2	X3	X4	X1	X2	X3	X4
1	7,9	7,2	5,7	6	7,2	7,2	6,0	8	10,0	9,8	7,1	7
2	9,0	7,2	7,0	5	8,7	7,3	5,7	5	9,2	9,0	5,8	6
3	6,6	7,8	6,9	6	6,2	6,8	5,8	8	7,9	6,8	6,4	6
РЕЙТИНГОВА СИСТЕМА												
1	5	5	3	4	5	5	3	5	5	5	5	5
2	5	5	5	4	5	5	3	4	5	5	3	4
3	3	5	5	4	3	4	3	5	5	4	3	4

За вхідними даними з табл. 6 видно, що проект № 20 має переважно вищі оціночні бали у порівнянні з 18 та 19 проектами, тому цей проект, найочевидніше, повинен займати перше місце у 3-му класі.

При рейтинговому оцінюванні лінгвістичні терми відповідають таким реальним балам:

«Погано» — 1 (X1); 1 (X2); 1 (X3); 3 (X4).

«Добре» — 3 (X1); 4 (X2); 3 (X3); 4 (X4).

«Гарно» — 5 (X1); 5 (X2); 5 (X3); 5 (X4).

Стосовно порівняння з реальними оцінками (рейтингова система) доцільно відзначити, що при рейтинговій системі дуже складно оцінювати місця проектів, які мають однакову кількість балів, тобто немає різниці між значеннями, що відповідають одному лінгвістичному терму. Складність оцінювання показана у табл. 6. Отже, рейтингова система класифікації проектів не гарантує завжди отримання однозначної класифікації (ранжування) проектів. Ця проблема зникає при класифікації проектів нечіткою експертною системою, зокрема, завдяки використанню показника компетентності експертів.

Якість отриманої класифікації інноваційних проектів можна підвищити за рахунок залучення до формування навчальної вибірки кваліфікованих експертів, оскільки представлена в експерименті навчальна вибірка сформована авторами роботи на основі дослідження предметної області вхідних даних та завдяки наданій інформації з відділу експертної оцінки науково-дослідних робіт НТУУ «КПІ».

ВИСНОВКИ

1. Запропоновано нечітку нейронну мережу для задачі класифікації проектів, поданих на конкурс в умовах неповної, нечіткої та якісної інформації.

2. Проведено дослідження розробленої системи стосовно задачі експертного оцінювання проектів, поданих на конкурс, та порівняння її з існуючою рейтинговою системою оцінювання проектів.

ЛІТЕРАТУРА

1. *Згуровский М.З.* Видение будущего — удел победителей // Зеркало недели. — 2001. — №25 (349).
2. *Згуровський М.З.* Сценарний аналіз як системна методологія передбачення // Системні дослідження та інформаційні технології. — 2002. — № 1. — С. 7–38.
3. *Панкратова Н.Д.* Математическое обеспечение задач технологического предвидения применительно к отрасли промышленности // Системні дослідження та інформаційні технології. — 2003. — № 1. — С. 26–33.
4. *Круглов В.В., Борисов В.В.* Принятие решений на основе нечетких моделей: примеры использования. — М.: Горячая линия. Телеком. — 2001. — 340 с.
5. *Зайченко Ю.П.* Основи проектування інтелектуальних систем. — Київ: Вид. дім «Слово», 2003. — 352 с.

Надійшла 08.10.2005