ЛЕКЦІЯ №7
Типи мереж.

1. Мережа Кохонена

Мережа розроблена Тойво Кохоненом на початку 1980-х рр. і принципово відрізняється від розглянутих вище мереж, оскільки використовує неконтрольоване навчання і навчальна множина складається лише із значень вхідних змінних.

Мережа розпізнає кластери в навчальних даних і розподіляє дані до відповідних кластерів. Якщо в подальшому мережа зустрічається з набором даних, несхожим ні з одним із відомих зразків, вона відносить його до нового кластеру. Якщо в даних містяться мітки класів, то мережа спроможна вирішувати задачі класифікації. Мережі Кохонена можна використовувати і в задачах, де класи є відомими - перевага буде у спроможності мережі виявляти подібність між різноманітними класами.

Мережа Кохонена має лише два прошарки: вхідний і вихідний (рис. 1.) її ще називають самоорганізовуваною картою. Елементи карти розташовуються в деякому просторі, як правило, двовимірному. Мережа Кохонена навчається методом послідовних наближень. У процесі навчання на входи подаються дані, але *мережа при цьому підлаштовується не під еталонне значення виходу, а під закономірності у вхідних даних*. Починається навчання з вибраного випадковим чином вихідного розташування центрів.

В процесі послідовної подачі на вхід мережі навчальних прикладів визначається найбільш схожий нейрон (той, у якого скалярний добуток ваг і поданого на вхід вектора є мінімальним). Цей нейрон оголошується переможцем і є центром при підлаштуванні ваг в сусідніх нейронів. Таке правило навчання передбачає "змагальне" (від «змагання») навчання з врахуванням відстані нейронів від "нейрона-переможця".

Навчання при цьому полягає не в мінімізації помилки, а в підлаштуванні ваг (внутрішніх параметрів нейронної мережі) для найбільшого збігу з вхідними даними.

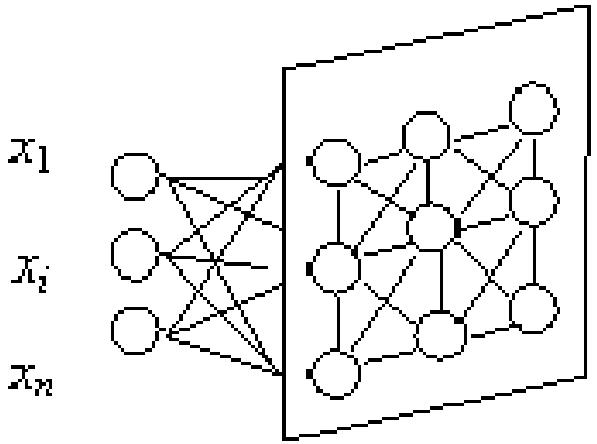


Рис. 1. Мережа Кохонена

Мережа Кохонена навчається методом послідовних наближень. Починаючи з випадковим чином обраного вихідного розташування центрів, алгоритм поступово поліпшується для кластеризації навчальних даних. Основний ітераційний алгоритм Кохонена послідовно проходить ряд епох, на кожній з яких обробляється один приклад з навчальної вибірки. Вхідні сигнали послідовно пред'являються мережі, при цьому бажані вихідні сигнали не визначаються. Після пред'явлення достатнього числа вхідних векторів синаптичні ваги мережі стають здатні визначити кластери. Ваги організуються так, що топологічно близькі вузли реагують до схожих вхідних сигналів.

В результаті роботи алгоритму центр кластера встановлюється в певній позиції, яка задовольняє кластеризовані приклади, для яких даний нейрон є "переможцем". В результаті навчання мережі необхідно визначити міру сусідства нейронів, тобто окіл нейрона-переможця, який представляє кілька нейронів, що оточують нейрон-переможець.

Для реалізації алгоритму необхідно визначити міру сусідства нейронів (околиця нейрона-переможця). На рис. 2 показані зони топологічного сусідства нейронів на карті ознак у різні моменти часу. NEj(t) – множина нейронів, що вважаються сусідами нейрона j у момент часу t. Зони сусідства зменшуються з часом.

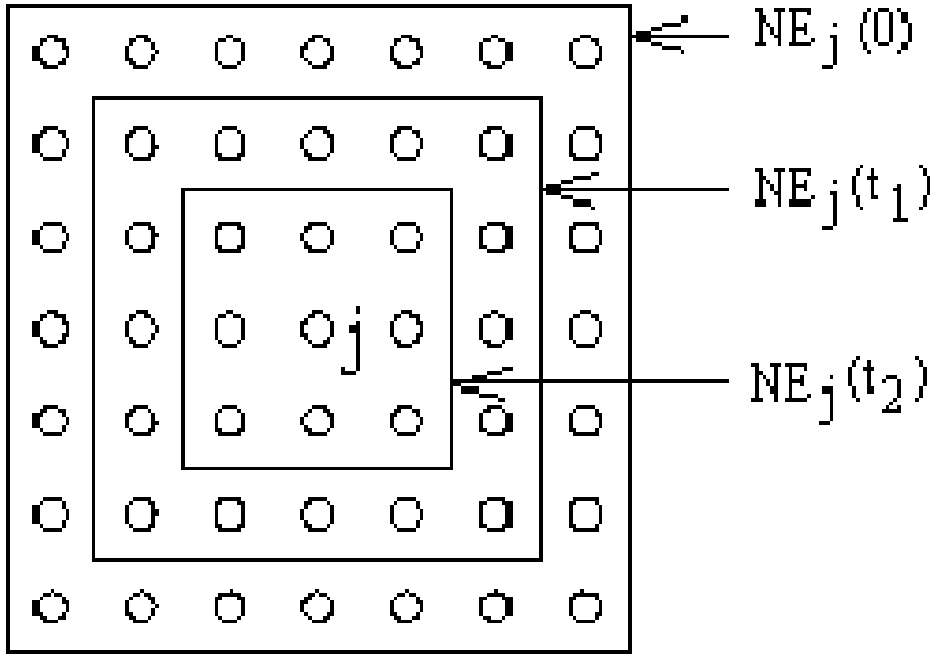


Рис. 2. Зони топологічного сусідства на карті ознак у різні моменти часу

Спочатку до околу належить велике число нейронів, далі її розмір поступово зменшується. Мережа формує топологічну структуру, в якій схожі приклади утворюють групи прикладів, які близько знаходяться на топологічної карті.

Алгоритм функціонування мережі Кохонена:

1. *Ініціалізація мережі.* Ваговим коефіцієнтам мережі надаються малі випадкові значення. Заються невеликі випадкові значення. Загальне число синаптичних ваг - M?N (див. рис. 1 Початкова зона сусідства показана на рис. 2.

2. *Пред'явлення мережі нового вхідного сигналу.*

3. *Обчислення відстані до всіх нейронів мережі:*
Відстані d_j від вхідного сигналу до кожного нейрона j визначаються за формулою:

$$d_j = \sum_{i=1}^{N} \left(x_i(t) \cdot w_{ij}(t)\right)^2$$

де x_i - *i-ий* елемент вхідного сигналу в момент часу t, $w_{ij}(t)$ - вага зв'язку від *i-го* елемента вхідного сигналу до нейрона j у момент часу t.

4. *Вибір нейрона з найменшою відстанню:*
Вибирається нейрон-переможець j^*, для якого відстань d_j найменше.

5. *Налаштування ваг нейрона j^* і його сусідів*:

Робиться налаштування ваг для нейрона j^* і всіх нейронів з його околу NE. Нові значення ваг:

$$w_{ij}(t+1)=w_{ij}(t)+r(t)(x_i(t)-w_{ij}(t))$$

де $r(t)$ - швидкість навчання, що зменшується з часом (додатне число, менше одиниці).

6. *Повернення до кроку 2.*

В алгоритмі використовується коефіцієнт швидкості навчання, який поступово зменшується, для тонкішої корекції в новій епосі. В результаті позиція центру встановлюється в певній позиції. Вона задовільним чином кластеризує приклади, для яких даний нейрон є переможцем.

Властивість топологічної впорядкованості досягається в алгоритмі за допомогою використання поняття околу. **Окіл** - це декілька нейронів, що оточують нейрон-переможець. Відповідно до швидкості навчання, розмір околу поступово зменшується, так, що спочатку до нього належить досить велике число нейронів (можливо вся карта), на самих останніх етапах окіл стає нульовим і складається лише з нейрона-переможця.

В алгоритмі навчання корекція застосовується не тільки до нейрона-переможця, але і до всіх нейронів з його поточного околу. В результаті такої зміни околу, початкові доволі великі ділянки мережі мігрують в бік навчальних прикладів. Мережа формує грубу структуру топологічного порядку, при якій схожі приклади активують групи нейронів, що близько знаходяться на топологічній карті.

З кожною новою епохою швидкість навчання і розмір околу зменшуються, тим самим всередині ділянок карти виявляються більш тонкі розходження, що зрештою призводить до точнішого налаштування кожного нейрона.

Часто навчання зумисне розбивають на дві фази: більш коротку, з великою швидкістю навчання і великих околів, і більш тривалу з малою швидкістю навчання і нульовими або майже нульовими околами.

Після того, як мережа навчена розпізнавати структури даних, її можна використовувати як засіб візуалізації при аналізі даних.

Області застосування. Кластерний аналіз, розпізнавання образів, класифікація.

Недоліки. Мережа може бути використана для кластерного аналізу маючи заздалегідь відоме число кластерів.

Переваги. Мережа Кохонена здатна функціонувати в умовах завад, тому що число кластерів фіксоване, ваги модифікуються повільно, налаштування ваг закінчується після навчання.

Мережа зустрічного поширення (CounterPropagation)

Роберт Хехт-Нильсен (Robert Hecht-Nielsen) розробив мережу Counterpropagation як засіб для об'єднання неконтрольованого шару Кохонена з контрольованим вихідним шаром. Мережа призначена для розв'язання складних класифікацій, при мінімізації числа нейронів і часу навчання. Навчання для мережі Counterpropagation схоже на мережі з квантуванням навчального вектора. Приклад мережі зображено на рис. 4. Односпрямована мережа CounterPropagation має три шари: вхідний шар, самоорганізована карта Кохонена і вихідний шар, що використовує правило "дельта" для зміни вхідних ваг з'єднань. Цей шар називають шаром Гроссберга.

Перша мережа Counterpropagation складалася з двонаправленого відображення між вхідним і вихідним шарами. Дані надходять на вхідний шар для генерації класифікації на вихідному шарі, вихідний шар по черзі приймає додатковий вхідний вектор і генерує вихідну класифікацію на вхідному шарі мережі. Через такий зустрічно-расповсюджений потік інформації мережа одержала свою назву. Багато розроблювачів використовують односпрямований варіант Counterpropagation, коли існує лише один шлях прямого поширення від вхідного до вихідного шару. У мережі зустрічного поширення об'єднані два алгоритми: самоорганізована карта Кохонена і зірка Гроссберга (Grossberg Outstar). Нейроні мережі, що поєднують різні нейропарадигми як будівельні блоки, більш близькі до мозку по архітектурі, чим однорідні структури. Вважається, що в мозку саме каскадні з'єднання модулів різної спеціалізації дозволяють виконувати необхідні обчислення.

Кожен елемент вхідного сигналу подається на всі нейрони шару Кохонена. Ваги зв'язків (wmn) утворюють матрицю W. Кожен нейрон шару Кохонена з'єднаний із усіма нейронами шару Гроссберга. Ваги зв'язків (vnp) утворюють матрицю ваг V. У процесі навчання мережі зустрічного поширення вхідні вектори асоціюються з відповідними вихідними векторами (двоїстими або аналоговими).

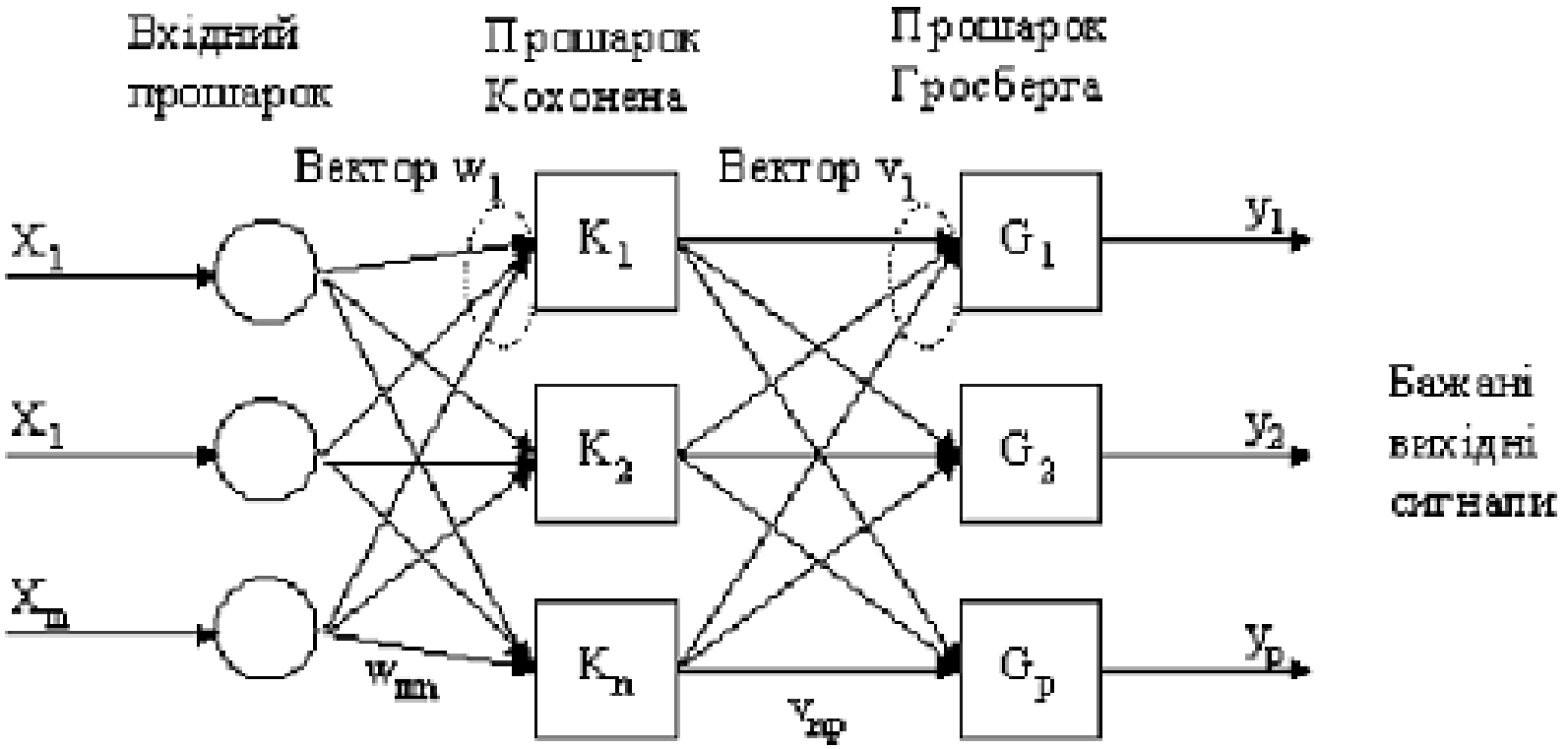


Рис. 4. Мережа зустрічного поширення без зворотних зв'язків

Після навчання мережа формує вихідні сигнали, що відповідають вхідним сигналам. Узагальнююча здатність мережі дає можливість одержувати правильний вихід, коли вхідний вектор неповний або перекручений.

Навчання мережі

У результаті самонавчання шар здобуває здатність розділяти несхожі вхідні вектори. Який саме нейрон буде активуватися при пред'явленні конкретного вхідного сигналу, заздалегідь важко передбачити. При навчанні шару Кохонена на вхід подається вхідний вектор і обчислюються його скалярні добутки з векторами ваг усіх нейронів. Скалярний добуток є мірою подібності між вхідним вектором і вектором ваг. Нейрон з максимальним значенням скалярного добутку вважається "переможцем" і його ваги підсилюються (ваговий вектор наближається до вхідного).

$$w_н = w_c + r(x - w_c)$$

де wн - нове значення ваги, що з'єднує вхідний компонент x з нейроном, що виграв, wc - попереднє значення ваги, r - коефіцієнт швидкості навчання, що спочатку зазвичай дорівнює 0,7 і може поступово зменшуватися в процесі навчання. Це дозволяє робити великі початкові кроки для швидкого грубого навчання і менші кроки при підході до остаточної величини.

Ймовірнісна нейрона мережа

Ймовірнісна нейрона мережа була розроблена Дональдом Спехтом (Donald Specht). Ця мережна архітектура була вперше представлена в двох статтях : "Ймовірнісні нейроні мережі для класифікації" (Probabilistic Neural Networks for Classification) 1988, "Відображення або асоціативна пам'ять і ймовірнісні нейроні мережі" (Mapping or Associative Memory and Probabilistic Neural Networks) 1990 р.

Виходи мережі можна інтерпретувати, як оцінки ймовірності приналежності елемента до визначеного класу. Ймовірнісна мережа учиться оцінювати функцію щільності ймовірності, її вихід розглядається як очікуване значення моделі в даній точці простору входів. Це значення зв'язане з щільністю ймовірності загального розподілу вхідних і вихідних даних. Задача оцінки щільності ймовірності відноситься до області байєсівської статистики. Звичайна статистика по заданій моделі показує, яка ймовірність того або іншого виходу (наприклад, на гральній кісті 6 очок буде випадати в середньому в одному випадку із шести).

Байєсівська статистика інтерпретує по іншому: правильність моделі оцінюється по наявним достовірним даним, тобто дає можливість оцінювати щільність ймовірності розподілу параметрів моделі по наявним даним. При рішенні задач класифікації можна оцінити щільність ймовірності для кожного класу, порівняти між собою ймовірності приналежності до різних класів і вибрати модель з параметрами, при яких щільність ймовірності буде більшою.

Оцінка щільності ймовірності в мережі заснована на ядерних оцінках. Якщо приклад розташований у даній точці простору, тоді в цій точці є визначена щільність ймовірності. Кластери з поруч розташованих точок, свідчать, що в цьому місці щільність ймовірності велика. Біля спостереження є більша довіра до рівня щільності, а в міру віддалення від нього довіра зменшується і прагне до нуля. У методі ядерних оцінок у точці, що відповідає кожному прикладові, міститься деяка проста функція, потім вони усі додаються й у результаті виходить оцінка для загальної щільності ймовірності. Частіше як ядерні функції беруть колоколоподібні функції (гауссовскі). Якщо є достатня кількість навчальних прикладів, такий метод дає гарні наближення до дійсної щільності ймовірності. Ймовірнісна мережа має три шари: вхідний, радіальний і вихідний. Радіальні елементи беруться по одному на кожен приклад. Кожний з них має гауссову функцію з центром у цьому прикладі. Кожному класові відповідає один вихідний елемент. Вихідний елемент з'єднаний лише з радіальними елементами, що відносяться до його класу і підсумовує виходи всіх елементів, що належать до його класу. Значення вихідних сигналів виходять пропорційно ядерним оцінкам ймовірності приналежності відповідним класам.

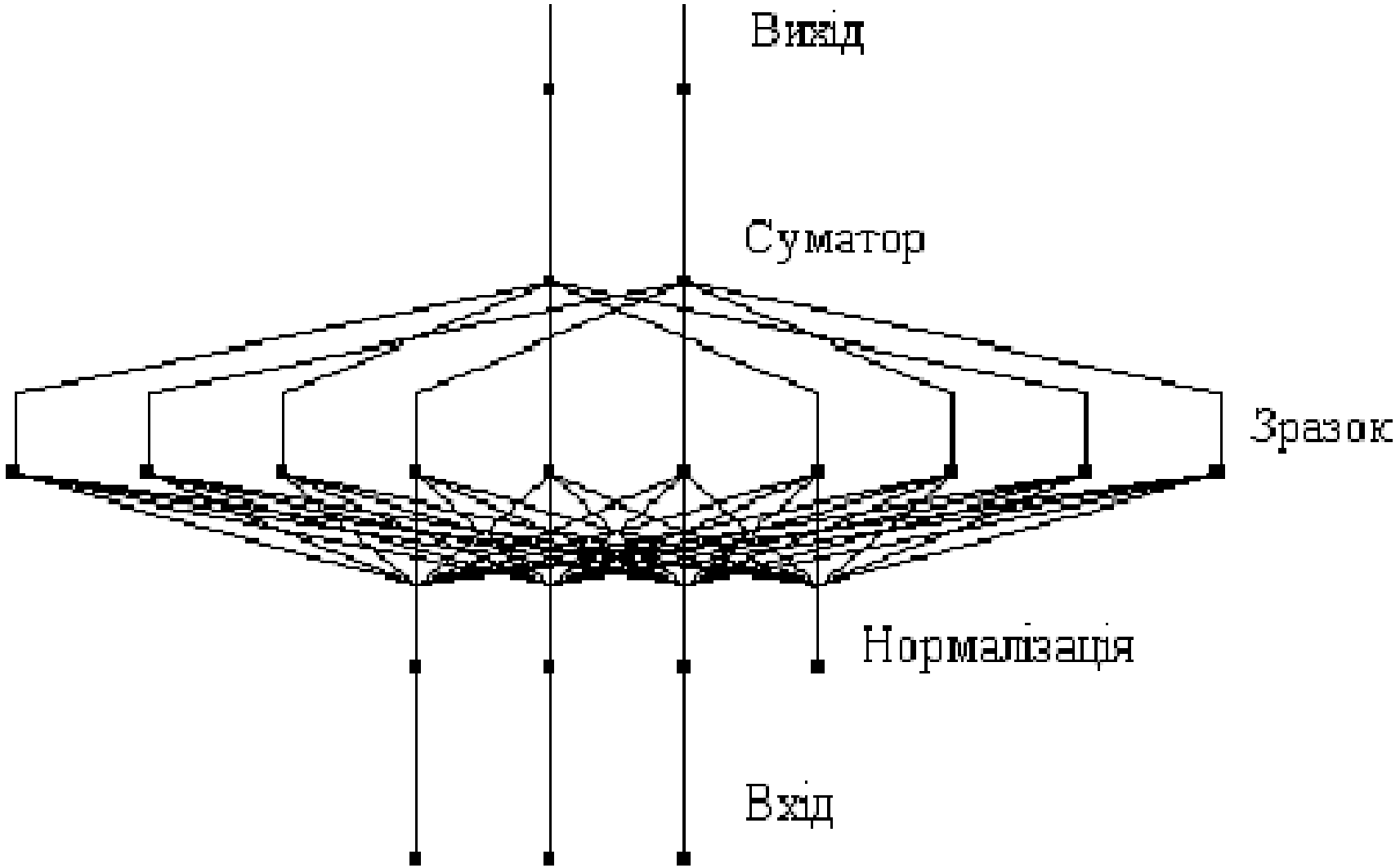


Рис. 5. Схема ймовірністної нейронної мережі.

Переваги. Навчання ймовірнисної нейронної мережі набагато простіше, ніж BackPropagation.

Недоліки. Недоліком мережі є її розмір, оскільки вона фактично містить у собі всі навчальні дані, вимагає багато пам'яті і може повільно працювати.

Модифікації. Базова модель ймовірнисної нейроної мережі має модифікації. Припустимо, що пропорції класів у навчальній множині відповідають їх пропорціям у всій досліджуваній множині (апріорна ймовірність). Наприклад, якщо серед усіх людей хворими є 2%, то в навчальній множині для мережі, що діагностує захворювання, хворих також повинне бути 2%. Якщо ж апріорні ймовірності відрізняються від пропорції в навчальній вибірці, мережа буде видавати невірний результат. Це можна врахувати, уводячи коригувальні коефіцієнти для різних класів.

Мережа Хопфілда

Мережа Хопфілда використовує три прошарки: вхідний, прошарок Хопфілда та вихідний прошарок. Кожен прошарок має однакову кількість нейронів. Виходи нейронів вхідного прошарку надходять до входів відповідних нейронів прошарку Хопфілда. Тут, зв'язки мають фіксовані вагові коефіцієнти. Виходи прошарку Хопфілда під'єднуються до входів всіх нейронів прошарку Хопфілда, за винятком самого себе, а також до відповідних елементів у вихідному прошарку. Під час навчання, мережа скеровує дані з вхідного прошарку до прошарку Хопфілда. Прошарок Хопфілда коливається, поки не буде завершена певна кількість циклів, і біжучий стан сигналів нейронів прошарку передається на вихідний прошарок. Цей стан відповідає образу, який буде запам'ятовано в мережі.

Навчання мережі Хопфілда вимагає, щоб навчальний образ був представлений на вхідному та вихідному прошарках одночасно. Рекурсивний характер прошарку Хопфілда забезпечує засоби корекції всіх ваг з'єднань. Для правильного навчання мережі відповідні пари "вхід-вихід" мають відрізнятися між собою.

Якщо мережа Хопфілда використовується як пам'ять, що адресується за змістом вона має два головних обмеження.

Число образів, що можна зберегти та точно відтворити є строго обмеженим. Якщо зберігається занадто багато образів, мережа може збігтись до нового неіснуючого образу, відмінному від всіх запрограмованих образів, або не збігтись взагалі. Межа ємності пам'яті для мережі приблизно 15% від числа нейронів у прошарку Хопфілда.

Якщо навчальні приклади є занадто подібними, прошарок Хопфілда може стати нестабільним. Зразок образу вважається нестабільним, якщо він застосовується за нульовий час і мережа збігається до деякого іншого образу з навчальної множини. Ця проблема може бути вирішена вибором навчальних прикладів більш ортогональних між собою.

Структурна схема мережі Хопфилда приведена на рис. 6

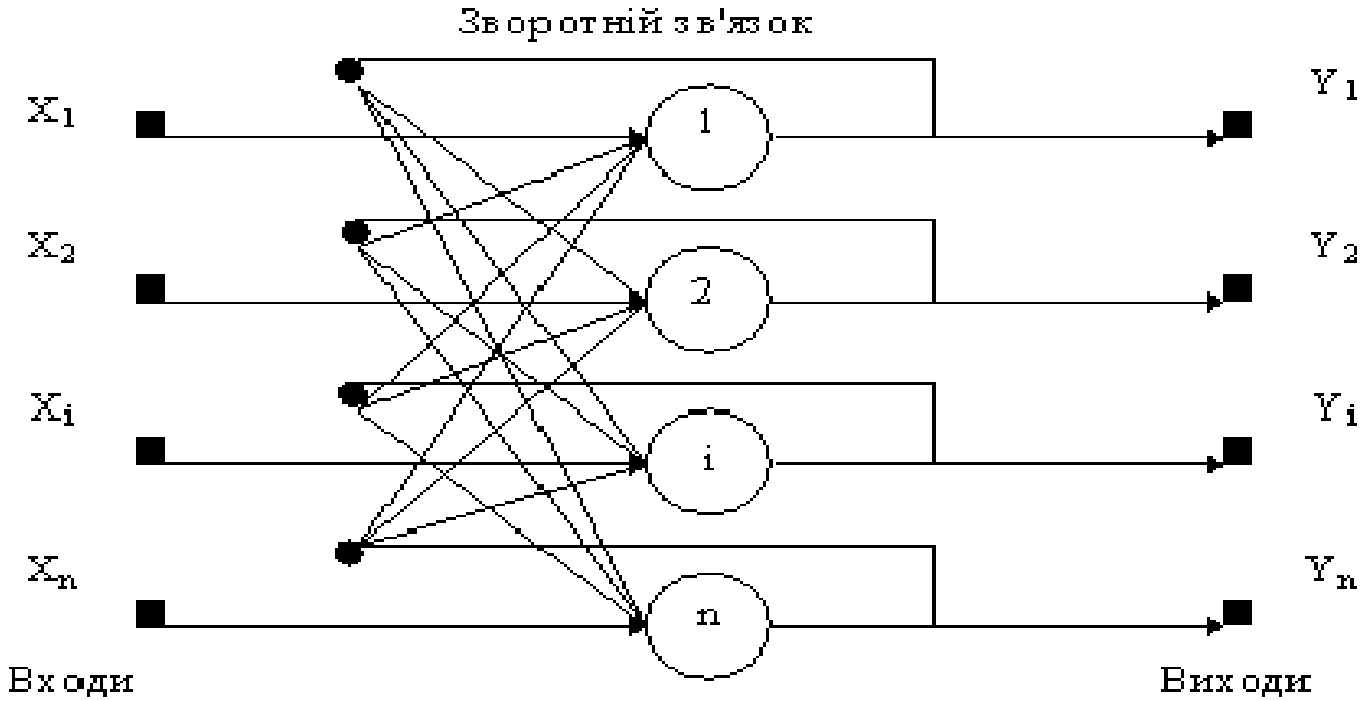


Рис.6. Структурна схема мережі Хопфилда

Для вирішення завдання асоціативної пам'яті є деякий набір двійкових сигналів (зображень, звукових оцифровок, інших даних, що описують об'єкти або характеристики процесів), який вважається зразковим. Мережа повинна вміти з зашумленого сигналу, поданого на її вхід, виділити ("пригадати" за частковою інформацією) відповідний зразок або "дати висновок" про те, що вхідні дані не відповідають жодному із зразків.

В загальному випадку, будь-який сигнал може бути описаний вектором $x_1, x_i, x_n..., n$ - число нейронів у мережі і величина вхідних і вихідних векторів. Кожний елемент x_i дорівнює або +1, або -1. Позначимо вектор, що описує k-ий зразок, через X_k, а його компоненти, відповідно, - x_{ik}, k=0, ..., m-1, m - число зразків. Якщо мережа розпізнає (або "пригадує") якийсь зразок на основі пред'явлених їй даних, її виходи будуть містити саме його, тобто $Y = X_k$, де Y - вектор вихідних значень мережі: y_1, y_i, y_n. У противному випадку, вихідний вектор не збігається з жодний зразковим.

Алгоритм функціонування мережі

1. На стадії ініціалізації мережі вагові коефіцієнти синапсів встановлюються таким чином:

$$w_{ij} = \begin{cases} \sum_{k=1}^{m} x_i^k x_j^k, & i \neq j \\ 0, & i = j \end{cases}$$

Тут *i* і *j* - індекси, відповідно, передсинаптичного і постсинаптичного нейронів; x_i^k *xik*, *xjk* - *i*-ий і *j*-ий елементи вектора *k*-ого зразка.

2. На входи мережі подається невідомий сигнал (t - номер ітерації). Його поширення безпосередньо встановлює значення виходів:

$y_i(0) = x_i$, $i = 0...n-1$,

тому позначення на схемі мережі вхідних сигналів у явному виді носить чисто умовний характер. Нуль у скобці справа від y_i означає нульову ітерацію в циклі роботи мережі.

3. Обчислюється новий стан нейронів

$$S(t+1) = \sum_{i=1}^{n} w_{ij} y_j(t), \quad j=0...n-1$$

і нові значення виходів

$$y_j(t+1) = f\left[S_j(t+1)\right]$$

де *f* - передатна функція у виді порогової, приведена на рис.7

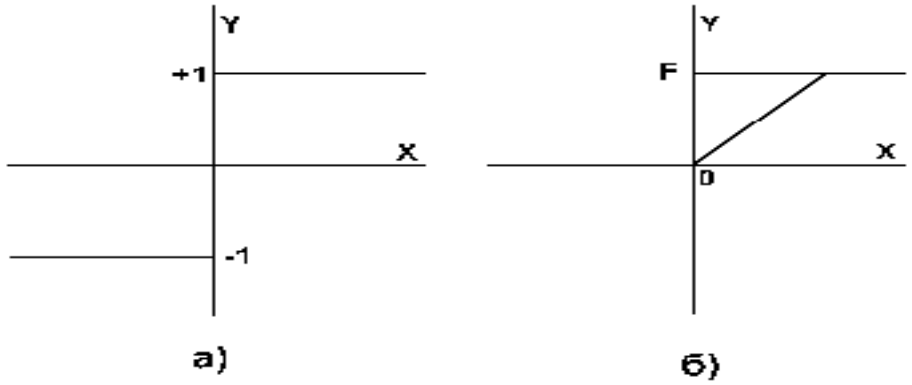


Рис.7. Порогові функції

4. Перевіряємо чи змінилися вихідні значення виходів за останню ітерацію. Якщо так – перехід до пункту 2, інакше (якщо виходи стабілізувались) – кінець. При цьому вихідний вектор являє собою зразок, що найкраще відповідає вхідним даним.

Іноді мережа не може провести розпізнавання і видає на виході неіснуючий образ. Це пов'язано з проблемою обмеженості можливостей мережі. Для мережі Хопфилда число збережених образів *m* не повинно перевищувати 0.15**n* (*n* – кількість нейронів вихідного прошарку) Крім того, якщо два образи А и Б сильно схожі, вони, можливо, будуть викликати в мережі перехресні асоціації, тобто пред'явлення на входи мережі вектора А призведе до появи на її виходах вектори Б и навпаки.

Мережа Хемінга.

Мережа Хемінга має три прошарки: вхідний прошарок з кількістю вузлів, скільки є окремих двійкових ознак; прошарок категорій (прошарок Хопфілда), з кількістю вузлів, скільки є категорій або класів; вихідний прошарок, який відповідає числу вузлів у прошарку категорій.

Мережа є простою архітектурою прямого поширення з вхідним рівнем, повністю під'єднаним до прошарку категорій. Кожен нейрон у прошарку категорій є зворотно під'єднаним до кожного нейрона у тому ж самому прошарку і прямо під'єднаним до вихідного нейрону. Вихід з прошарку категорій до вихідного прошарку формується через конкуренцію.

Навчання мережі Хемінга є подібним до методології Хопфілда. На вхідний прошарок надходить бажаний навчальний образ, а на виході вихідного прошарку надходить значення бажаного класу, до якого належить вектор. Вихід містить лише значення класу до якої належить вхідний вектор. Рекурсивний характер прошарку Хопфілда забезпечує засоби корекції всіх ваг з'єднань.

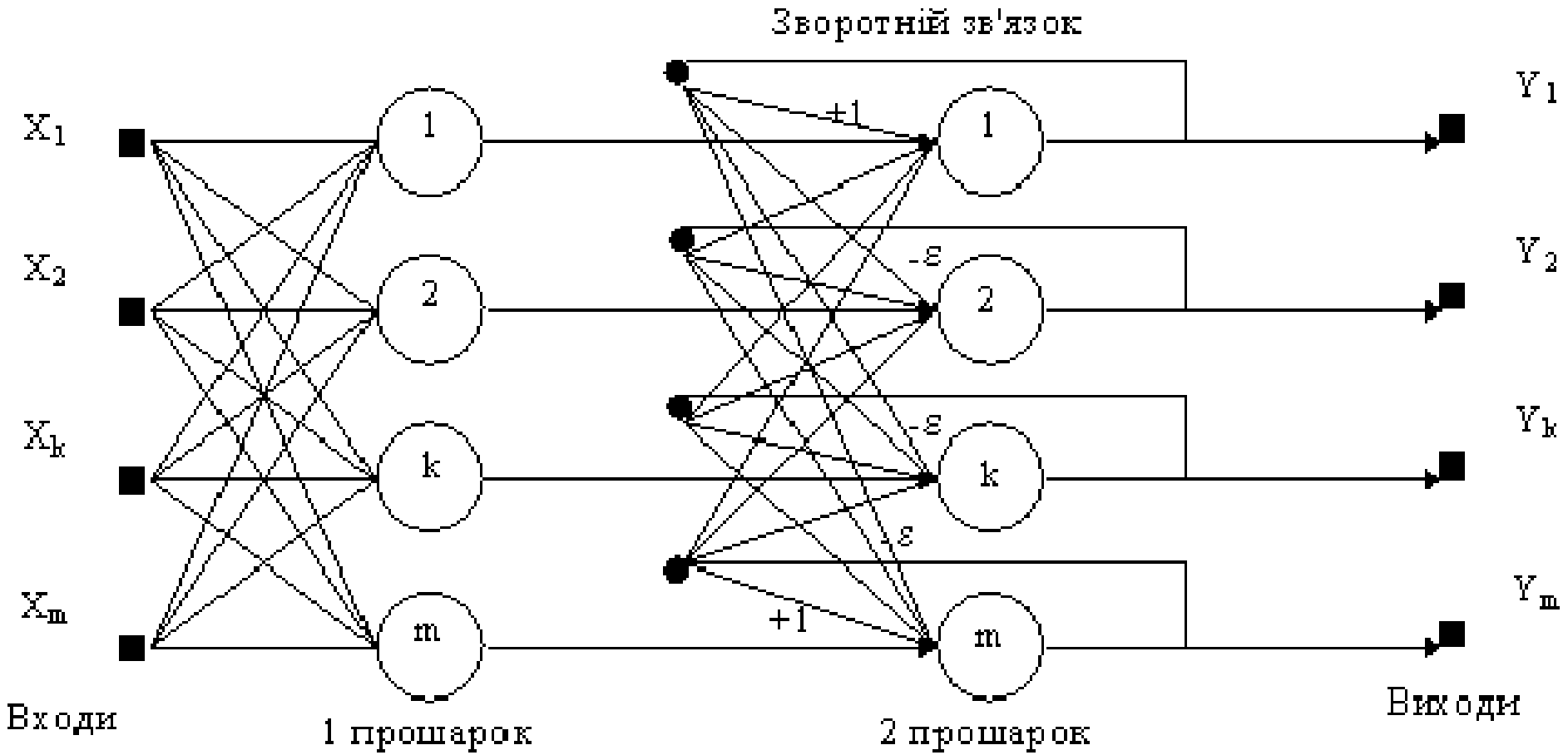

Зворотній зв'язок
X1
X2
Xk
Xm
1
2
k
m
+1
-ε
-ε
ε
+1
1
2
k
m
Y1
Y2
Yk
Ym
Входи
1 прошарок
2 прошарок
Виходи

Рис.8.

Алгоритм функціонування мережі Хемінга

1. На стадії ініціалізації ваговим коефіцієнтам першого прошарку і порогу передатної функції присвоюються такі значення:

$W_{ik}=x_i^k/2$, i=0...n-1, k=0...m-1

$b_k = n / 2$, k = 0...m-1

Тут x_i^k - *i*-ий елемент *k*-ого зразка.

Вагові коефіцієнти гальмуючих синапсів у другому прошарку беруть рівними деякій величині $0 < v < 1/m$. Синапс нейрона, пов'язаний із його ж виходом має вагу +1.

2. На входи мережі подається невідомий вектор x_1, x_i, x_n ...Розраховуються стани нейронів першого прошарку (верхній індекс у скобках указує номер прошарку):

$$y_j^{(1)} = S_j^{(1)} = \sum_{i=1}^{n} w_{ij} x_i + b_j \quad , j=0...m-1$$

Після цього отримання значення ініціалізують значення виходів другого прошарку:

$y_j^{(2)} = y_j^{(1)}$, j = 0...m-1

3. Обчислюються нові стани нейронів другого прошарку:

$$S_j^{(2)}(t+1) = y_j(t) - \varepsilon \sum_{k=1}^{m} y_k^{(2)}(t),\ k \neq j,\ j = 1...m$$

і значення їх виходів:

$$y_j^{(1)}(t+1) = f\left[S_j^{(2)}(t+1)\right]$$

Передатна функція *f* має вид порога, причому величина *b* повинна бути достатньо великою, щоб будь-які можливі значення аргументу не призводили до насичення.

4. Перевіряється, чи змінилися виходи нейронів другого прошарку за останню ітерацію. Якщо так - перейти до кроку 3. Інакше - кінець.

Мережа з двонаправленою асоціативною пам’ятю

Ця мережна моделі була розроблена Бартом Козко (Bart Kosko) і розширює модель Хопфілда. Множина парних образів учиться по образах, що представлені як біполярні вектори. Подібно мережі Хопфілда, коли представляється зашумлена версія одного образа, визначається найближчий образ, асоційований з ним. На рис. 9 показаний приклад двонаправленої асоціативної пам'яті.

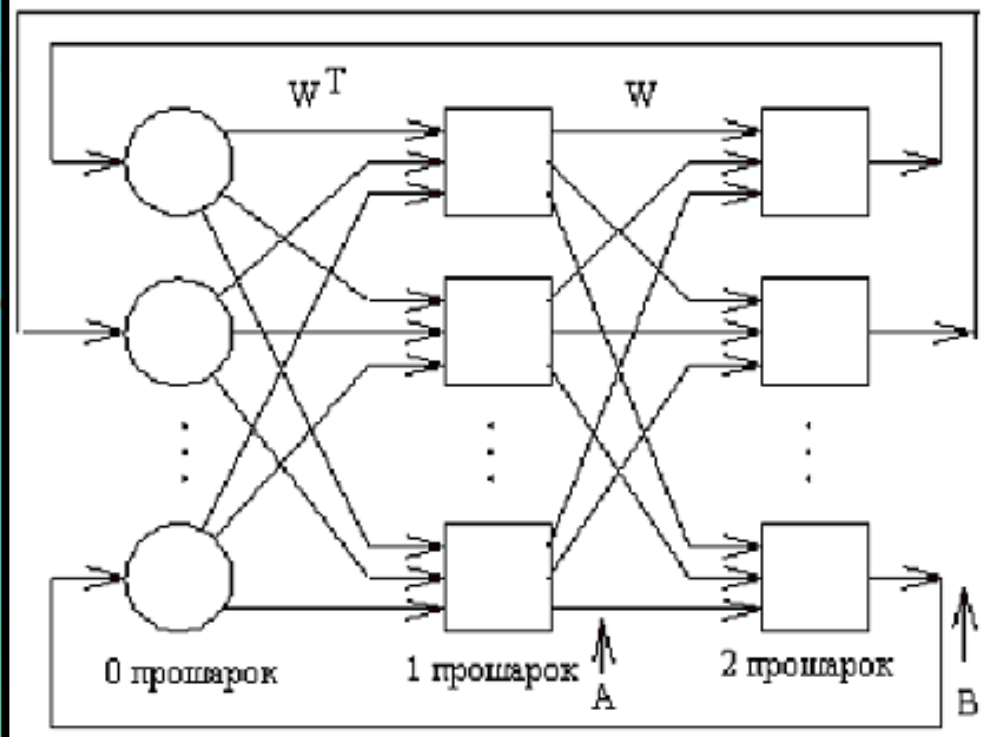


Рис. 9. Двонаправлена асоціативна пам'ять

Вхідний вектор A обробляється матрицею ваг W мережі, у результаті чого продукується вектор вихідних сигналів мережі B. Вектор B обробляється транспонованою матрицею WT ваг мережі, що продукує сигнали, які представляють новий вхідний вектор A. Цей процес повторюється доти, поки мережа не досягне стабільного стану, у якому ні вектор A, ні вектор B не змінюються.

Нейрони в шарах 1 і 2 функціонують, як і в інших парадигмах, обчислюючи суму зважених входів і значення передатної функції F:

$$b_j = F\sum_i a_i W_{ij}$$

або у векторній формі:

$$B=F(AW)$$

де B- вектор вихідних сигналів нейронів шару 2, A - вектор вихідних сигналів нейронів шару 1, W - матриця ваг зв'язків між шарами 1 і 2, F - передатна функція.

Аналогічно

$$A=F(BW^T),$$

де W^T є транспозицією матриці W.

Як передатну функцію використовується експонентна сігмоїда. Шар 0 не виконує обчислень і не має пам'яті. Він служить лише засобом розподілу вихідних сигналів шару 2 до елементів матриці WT

. Формула для обчислення значень синаптичних ваг:

$$W = \sum_j A_j^T B_j$$

де A_j і B_j - вхідні і вихідні сигнали навчальної множини.

Вагова матриця обчислюється як сума добутків усіх векторних пар навчальної множини. Системи зі зворотним зв'язком мають тенденцію до коливань. Вони можуть переходити від стану до стану, ніколи не досягаючи стабільності. Доведено, що двонаправлена асоціативна пам'ять безумовно стабільна при будь-яких значеннях ваг мережі.