

ДКЧП перевагу у численних застосуваннях над альтернативними РНМ, прихованими марковськими моделями та іншими методами навчання послідовностей.

Мережа ДКЧП складається з шарів, серед яких містяться вузли ДКЧП.

Вузол ДКЧП – це вузол РНМ, який виділяється запам'ятовуванням значень для довгих, або коротких проміжків часу. Це забезпечується завдяки тому, що вузол не використовує функції активації в межах своїх рекурентних складових. Таким чином, значення, що зберігається, не розплищується ітеративно з плином часу, і член градієнту або штраф не має схильності розмиватися, коли для його навчання застосовується зворотне поширення у часі.

Вузли ДКЧП часто втілюють у блоках, які містять декілька вузлів ДКЧП. Така конструкція є типовою для «глибинних» БНМ, і сприяє реалізаціям на паралельному апаратному забезпеченні.

Блоки ДКЧП містять три або чотири вентиля, які вони використовують для керування плином інформації до або з їхньої пам'яті. Ці вентиля реалізують із застосуванням логістичної функції для обчислення значень між 0 та 1. Для часткового дозволу або заборони плину інформації до або з цієї пам'яті застосовується множення на це значення.

Наприклад, вхідний вентиль керує мірою, до якої нове значення входить до пам'яті. Забувальний вентиль керує мірою, до якої значення залишається в пам'яті. А вихідний вентиль керує мірою, до якої значення в пам'яті використовується для обчислення активування виходу блоку. У деяких втіленнях вхідний та забувальний вентиля об'єднують в один. Ідея їхнього об'єднання полягає в тому, що час забувати настає тоді, коли з'являється нове значення, варте запам'ятовування.

Схема простого блоку ДКЧП зображена на рис. 5.2. Тут для блоку позначено: i – вхідні, o – вихідні та f – забувальні ворота. Кожне з цих воріт може розглядатися як нейрон БНМ: тобто вони обчислюють функцію активації зваженої суми, а i_t , o_t та f_t являють собою активації відповідно вхідних, вихідних та забувальних воріт у момент часу t . Три стрілки, що виходять з комірки пам'яті

інформації), i_t – вектор вхідного вентиля (вага отримання нової інформації), o_t – вектор вихідного вентиля (кандидатність на вихід); функції активації: a_g – сигмоїдна функція, a_c – гіперболічний тангенс, a_h – гіперболічний тангенс (але практично рекомендується використовувати лінійну функцію: $a_h(x)=x$). У наведених нижче рівняннях кожна змінна курсивом у нижньому регістрі подає вектор, що має розмір, який дорівнює числу вузлів ДКЧП у блоці. Початкові значення: $c_0 = 0, h_0 = 0$.

Традиційна ДКЧП із забувальними вузлами описується як:

$$\begin{aligned} f_t &= a_g(W_f x_t + U_f h_{t-1} + b_f), \\ i_t &= a_g(W_i x_t + U_i h_{t-1} + b_i), \\ o_t &= a_g(W_o x_t + U_o h_{t-1} + b_o), \\ \tilde{c}_t &= a_c(W_c x_t + U_c h_{t-1} + b_c), \\ c_t &= f_t \circ c_{t-1} + i_t \circ \tilde{c}_t, \\ h_t &= o_t \circ a_h(c_t). \end{aligned}$$

Вічкова ДКЧП із забувальними вентилями описується як:

$$\begin{aligned} f_t &= a_g(W_f x_t + U_f c_{t-1} + b_f), \\ i_t &= a_g(W_i x_t + U_i c_{t-1} + b_i), \\ o_t &= a_g(W_o x_t + U_o c_{t-1} + b_o), \\ c_t &= f_t \circ c_{t-1} + i_t \circ a_c(W_c x_t + b_c), \\ h_t &= o_t \circ a_h(c_t). \end{aligned}$$

Згорткова ДКЧП описується як:

$$\begin{aligned} f_t &= a_g(W_f^* x_t + U_f^* h_{t-1} + V_f \circ c_{t-1} + b_f), \\ i_t &= a_g(W_i^* x_t + U_i^* h_{t-1} + V_i \circ c_{t-1} + b_i), \\ o_t &= a_g(W_o^* x_t + U_o^* h_{t-1} + V_o \circ c_{t-1} + b_o), \\ c_t &= f_t \circ c_{t-1} + i_t \circ a_c(W_c^* x_t + U_c^* h_{t-1} + b_c), \\ h_t &= o_t \circ a_h(c_t), \end{aligned}$$

де * – оператор згортки.

Єдині ваги, що є в блоці ДКЧП W та U , використовуються для спрямування дії вентилів. Ці ваги застосовуються між значеннями, які надходять до блоку (включно з вхідним вектором x_t та виходом з попереднього моменту часу h_{t-1} та кожним із вентилів. Отже, блок ДКЧП визначає, яким чином підтримувати свою пам'ять як функцію від цих значень, і

тренування ваг блока ДКЧП спричиняє його навчання такої функції, яка мінімізує втрати.

Тренування ДКЧП-мереж здійснюють на основі техніки зворотного поширення помилки у часі. Також використовують еволюційні методи.

5.4 Гібридні глибинні мережі

Утворити ГНМ можливо шляхом певного поєднання більш простих нейромереж. Розглянемо кілька найбільш популярних класів гібридних ГНМ.

Генеративні змагальні мережі (ГЗМ, generative adversarial networks, GANs) – це клас ГНМ, що використовуються в навчанні без учителя, реалізовані системою двох штучних нейронних мереж, які змагаються одна з одною в рамках гри з нульовою сумою.

Схему ГЗМ зображено на рис. 5.3.

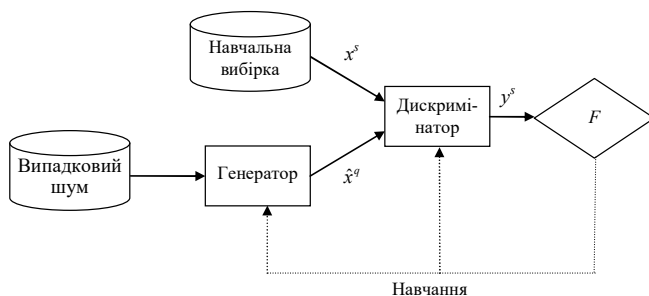


Рисунок 5.3 – Схема ГЗМ

Одна мережа – *генератор* генерує екземпляри-кандидати, а інша мережа – *дискриміратор* оцінює їх. Як правило, генератор навчається будувати відповідності з латентного (прихованого) простору до певного розподілу даних, тоді як дискримінаційна мережа розрізняє представників x^s справжнього розподілу даних (навчальної вибірки) та кандидатів \hat{x}^g , вироблених генератором. Метою генератора як тренувальної мережі є збільшення частоти помилок дискримінаційної мережі (тобто «обдурити» дискриміратор шляхом створення нових синтезованих екземплярів, які повинні походити на представників справжнього розподілу даних).

На практиці заздалегідь відомий набір даних використовують як початкові навчальні данні для дискримінатора. Навчання дискримінатора передбачає забезпечення його зразками з набору даних, доки він не досягне певного рівня точності F . Зазвичай генератор на початку отримує випадково відібрані дані із заздалегідь визначеного латентного простору. Після цього екземпляри, синтезовані генератором, оцінюються дискримінатором.

Генератор, як правило, є деконволюційною НМ, а дискримінатор – ЗНМ.

Для навчання обох мереж застосовують техніку зворотного поширення помилки.

Сіамська нейронна мережа (siamese neural network, twin neural network) – це різновид ГНМ, яка використовує однакові ваги при одночасній роботі з двома різними вхідними векторами для обчислення вихідних векторів, що підлягають порівнянню.

Часто один з вихідних векторів попередньо обчислюється, утворюючи таким чином базу, з якою порівнюють інший вихідний вектор. Це схоже на локально-чуттєве хешування (тобто порівняння замість багатомірних оригінальних екземплярів їхніх гешів – свого роду "відбитків пальців" або описів-портретів невеликої розмірності).

Сіамська мережа складається з двох однакових НМ, кожна з яких приймає один з двох векторів вхідних даних. Виходи останніх шарів цих двох мереж подаються на контрастуючу функцію втрат, яка обчислює схожість між двома векторами вхідних даних. Узагальнену схему сіамської НМ зображено на рис. 5.4. Тут x^p , x^q – два різні екземпляри, подані на входи першої (net_1) та другої (net_2) НМ, відповідно. Мережі net_1 та net_2 на виходах видають значення y_1 та y_2 , які подаються на відповідні входи блоку функції втрат L . Цей блок обчислює вихід мережі y .

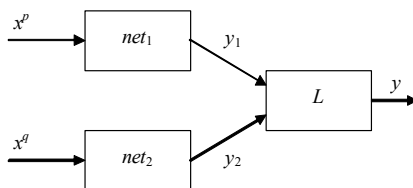


Рисунок 5.4 – Схема сіамської НМ


Навчання в сіамських мережах може бути здійснено з триплетними (трійковими) втратами або контрастними втратами. Для навчання за допомогою триплетних втрат базовий вектор порівнюється з позитивним вектором та негативним вектором. Негативний вектор змусить мережу навчатися, тоді як позитивний вектор буде діяти як регуляризатор. Для навчання за допомогою контрастних втрат необхідно запровадити зменшення ваги для регулювання ваг мережі або подібну операцію, наприклад, нормалізацію.

? 5.5 Контрольні питання

1. Що таке глибинне навчання?
2. Що таке глибинна нейронна мережа?
3. Порівняйте мілкі НМ та ГНМ.
4. Що таке згорткова нейронна мережа?
5. Наведіть схему згорткової нейромережі.
6. Опишіть структуру та функціонування згорткової нейромережі.
7. Для чого призначений та як функціонує згортковий шар?
8. Що таке згортка?
9. Для чого призначений та як функціонує агрегувальний шар?
10. Що таке максимізаційне агрегування (max pooling)?
11. Що таке агрегування областей інтересу (RoI pooling)?
12. Що таке усереднювальне агрегування (average pooling)?
13. Що таке L^2 -нормове агрегування (L^2 -norm pooling)?
14. Для чого призначений та як функціонує шар зрізаних лінійних вузлів (ReLU)?
15. Для чого призначений та як функціонує згладжувальний шар (flatten layer)?
16. Для чого призначений та як функціонує повноз'єднаний шар (fully connected layer)?
17. Для чого призначений та як функціонує шар втрат (loss layer)?
18. Опишіть основні відмінності ЗНМ від БНМ.
19. Що таке шар об'єднання?
20. Що таке мережа довгої короткочасної пам'яті (LSTM)?

21. Що таке вузол ДКЧП?
22. Зобразіть та опишіть схему простого блоку ДКЧП?
23. Наведіть математичний опис функціонування ДКЧП із забувальними вузлами.
24. Наведіть математичний опис функціонування вічкової ДКЧП із забувальними вентилями.
25. Наведіть математичний опис функціонування згорткової ДКЧП.
26. Які Ви знаєте гібридні глибинні нейромережі?
27. Як утворюють гібридні глибинні нейромережі?
28. Що таке генеративна змагальна мережа (GAN)?
29. Зобразіть схему генеративної змагальної нейромережі.
30. Що таке мережа-генератор?
31. Що таке мережа-дискримінатор?
32. Що таке сіамська нейронна мережа?
33. Зобразіть схему сіамської нейромережі.
34. Як навчають глибинні нейромережі?

5.6 Практичні завдання

 *Завдання 1.* Підготувати реферат на одну з таких тем.

1. Архітектури згорткових нейромереж.
2. Методи навчання згорткових нейромереж.
3. Методи навчання мереж LSTM.
4. Архітектури гібридних глибинних нейромереж.
5. Методи навчання змагальних глибинних нейромереж.
6. Застосування глибинних нейромереж.



Завдання 2. Емуляція та навчання глибинних мереж.

1. Обрати парадигму глибинної мережі, програмний засіб, що її реалізує та практичну задачу (навчальну вибірку). За допомогою обраного засобу побудувати нейро модель для відповідної задачі. Перевірити працездатність моделі на даних, що не використовувалися у навчанні. Зробити висновки щодо швидкості побудови, роботи, точності відповідної парадигми глибинних мереж.

2. Повторити п. 1 для іншої парадигми або програмного засобу. Порівняти отримані результати.