

УДК 004.093

**МАТЕМАТИЧНА БАЗА ДЛЯ СТВОРЕННЯ ГІБРИДНОЇ
НЕЙРОННОЇ МЕРЕЖІ ДЛЯ РЕКОМЕНДАЦІЙ
З ГЛИБОКИМ НАВЧАННЯМ**

Васильєв Р.Р.

Науковий керівник – к.т.н., доц. Кіношенко Д.К.

Харківський національний університет радіоелектроніки, каф. ІНФ,
м. Харків, Україна

e-mail: ruslan.vasyliiev@nure.ua

This work is devoted to the description of the mathematical framework that is needed to create a hybrid neural network for recommendations with deep learning. This approach to recommendations combines collaborative filtering and content processing of those elements on which training will be performed. The paper considers the mathematical side of the model which uses a loss function for training, a stochastic gradient descent (SGD) method and back propagation to adapt the model to minimize the loss. All these aspects described by mathematical formulas allow to write and train a neural network for recommendations, which can accommodate many parameters for recommendations and can be written in any platform and programming language.

У сучасному світі, де кількість інформації зростає експоненціально, величезне значення набуває ефективність її обробки та аналізу. Це особливо стосується галузей, де потрібно пропонувати персоналізовані рекомендації користувачам або клієнтам, як у сферах електронної комерції, соціальних мереж, стрімінгових сервісів тощо. Гібридні нейронні мережі, що використовують методи глибокого навчання, представляють собою передові технології в області рекомендаційних систем, здатні ефективно аналізувати великі обсяги даних та виявляти складні зв'язки між ними. Рекомендаційні системи стали невід'ємною частиною багатьох сучасних інтернет-платформ, допомагаючи користувачам знаходити релевантний контент серед нескінченної кількості можливостей. Використання гібридних нейронних мереж дозволяє значно підвищити якість рекомендацій за рахунок комбінування кількох типів даних та алгоритмів в одній моделі.

Гібридна нейронна мережа для рекомендацій поєднує в собі кілька підходів, зокрема колаборативну фільтрацію, обробку контенту та глибоке навчання, щоб врахувати різноманіття контекстів користувачів та предметів. Використання глибокого навчання дозволяє автоматично виявляти складні неявні залежності між характеристиками користувачів та предметів, що значно покращує якість рекомендацій.

Одним із ключових елементів навчання нейронних мереж є функція втрат, яка вимірює розбіжність між прогнозованими та реальними даними.

Для рекомендаційних систем часто використовується функція втрат квадратичної помилки:

$$L(Y, \hat{Y}) = \frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2,$$

де L – це функція втрат; Y – реальні рейтинги або вподобання; \hat{Y} – прогнозовані рейтинги; N – кількість спостережень.

Для оптимізації параметрів моделі часто використовується метод стохастичного градієнтного спуску (SGD) або його варіації, такі як Adam. Основна ідея полягає в оновленні кожного параметра θ в напрямку, протилежному градієнту функції втрат:

$$\theta_{new} = \theta_{old} - \eta \cdot \nabla_{\theta} L(Y, \hat{Y}),$$

де η – швидкість навчання; $\nabla_{\theta} L$ – градієнт функції втрат по параметру θ .

Виходячи з цього, для створення системи рекомендацій потрібно підготувати дані для навчання, створити модель нейронної мережі та обучити її. Для підготовки даних гарною практикою є використання готових наборів даних, які створюють наукові дослідницькі спільноти для відслідковування статистики. Обробка повинна містити у собі об'єднання різних наборів даних по одному аспекту, видалення непотрібних даних та заповнення порожніх місць.

Нейронна мережа будується з вузлів (нейронів), які організовані в шари. Вихід кожного нейрона є функцією від суми вагованих входів плюс зсув (bias). Таким чином ми повинні підлаштувати функції у вузлах таким чином щоб вони допомогли нейромережі надати рекомендації:

$$a^{[l+1]} = \sigma(W^{[l]}a^{[l]} + b^{[l]}),$$

де $a^{[l]}$ – активації нейронів на шарі l ; $W^{[l]}$, $b^{[l]}$ – ваги зсуви шару l відповідно; σ – функція активації.

Для задач рекомендацій можна використовувати функцію втрат, наприклад, крос-ентропію для задач класифікації або MSE для задач регресії:

$$\mathcal{L}(y, \hat{y}) = - \sum_{i=1}^n y_i \log \log(y_i) + (1 - y_i) \log \log(1 - y_i),$$

де y_i – істинне значення для i -го прикладу; \hat{y}_i – передбачення моделі для i -го прикладу; n – кількість прикладів у наборі даних.

Зворотне поширення помилки (backpropagation) використовується для обчислення градієнтів функції втрат по вагам та зсувам моделі, щоб адаптувати модель з метою мінімізації втрат:

$$\partial\Omega/\partial W^{[l]} = \frac{1}{m} \Delta W^{[l]},$$
$$\partial\Omega/\partial b^{[l]} = \frac{1}{m} \Delta b^{[l]},$$

де $\Delta W^{[l]}$, $\Delta b^{[l]}$ – градієнти функції втрат відносно ваг та зсувів шару; l , m – кількість прикладів у наборі даних.

Використовуючи зазначені формули можна легко створювати системи рекомендацій для будь якої доменної області, підготувавши заздалегідь різноманітні дані, створивши модель нейронної мережі, та провівши навчання зі зворотним поширенням.

Список використаних джерел:

1. Goodfellow, I., Bengio, Y., & Courville, A. (2016). Deep Learning. MIT Press.
2. Zhang, S., Yao, L., Sun, A., & Tay, Y. (2019). Deep Learning Based Recommender System: A Survey and New Perspectives. ACM Computing Surveys, 52(1), 1–38.
3. Rendle, S., Krichene, W., Zhang, L., & Anderson, J. (2020). Neural Collaborative Filtering vs. Matrix Factorization Revisited. In Proceedings of the 14th ACM Conference on Recommender Systems (RecSys '20). ACM, New York, NY, USA, 240–248.