

NEYRON TO'RLAR VA GENETIK ALGORITMLAR

Onarkulov Maksadjon Karimberdiyevich*Farg'ona davlat universiteti amaliy matematika va informatika
kafedrası dotsenti Gmail: maxmaqsad@gmail.com***Tojimamatov Israiljon Nurmamatovich***Farg'ona davlat universiteti amaliy matematika va informatika
kafedrası katta o'qituvchisi Gmail: isik80@mail.ru***Yusupov Mirsaid Abdulaziz o'g'li***Farg'ona davlat universiteti amaliy matematika va informatika
kafedrası o'qituvchisi Gmail: mirsaidbeky@gmail.com***Nabiyev Asadbek Akmaljon o'g'li***Farg'ona davlat universiteti 3-kurs talabasi
Gmail: nabiyevasadbek0808@gmail.com*

Annotatsiya. Ushbu maqolada neyron tarmoqlar va genetik algoritmlar mavzusi keng tahlil qilingan. Neyron tarmoqlarning ishlash prinsiplari, arxitekturalari va ularning sun'iy intellekt tizimlarida qo'llanilishi ko'rib chiqilgan. Shuningdek, genetik algoritmlarning evolyutsion jarayonlar va optimallashtirishda qanday ishlashi muhokama qilingan. Neyron tarmoqlar va genetik algoritmlarning o'zaro integratsiyasi hamda ushbu yondashuvlarning texnologik rivojlanish jarayonlariga qanday ta'sir ko'rsatayotgani haqida fikr yuritilgan.

Kalit so'zlar: neyron tarmoqlar, genetik algoritmlar, sun'iy intellekt, mashinani o'rganish, optimallashtirish, evolyutsion hisoblash.

Annotation. This article provides an extensive analysis of neural networks and genetic algorithms. It examines the working principles, architecture, and applications of neural networks in artificial intelligence systems. Additionally, the use of genetic algorithms in evolutionary processes and optimization is discussed. The integration of neural networks and genetic algorithms is explored, along with their impact on technological advancements.

Key words: neural networks, genetic algorithms, artificial intelligence, machine learning, optimization, evolutionary computing.

Аннотация. В данной статье проводится детальный анализ нейронных сетей и генетических алгоритмов. Рассматриваются принципы работы, архитектура и применение нейронных сетей в системах искусственного интеллекта. Также обсуждается использование генетических алгоритмов в эволюционных процессах и оптимизации. Особое внимание уделено интеграции нейронных сетей и генетических алгоритмов, а также их влиянию на технологические инновации.

Ключевые слова: нейронные сети, генетические алгоритмы, искусственный интеллект, машинное обучение, оптимизация, эволюционные вычисления.

Axborot texnologiyalari va sun'iy intellekt sohasidagi tezkor rivojlanish yangi matematik va algoritmik yondashuvlarni talab qilmoqda. Neyron tarmoqlar va genetik algoritmlar

zamonaviy hisoblash tizimlarining muhim komponentlari hisoblanadi. Neyron tarmoqlar ma'lumotlarni qayta ishlash va mustaqil ravishda o'rganish imkonini beruvchi sun'iy intellekt tizimlarining poydevori bo'lib, ulardan foydalanish texnologik jarayonlarni optimallashtirishda keng ko'lamda qo'llanilmoqda. Shu bilan birga, genetik algoritmlar evolyutsion jarayonlarga asoslangan yondashuv bo'lib, tabiiy tanlanish va genetik mutatsiyalarni hisoblash algoritmlariga qo'llanilib, murakkab muammolarni hal qilish uchun ishlatiladi.

Ushbu maqolada neyron tarmoqlar va genetik algoritmlar to'g'risidagi asosiy bilimlar, ularning ishlash mexanizmlari va o'zaro aloqasi tahlil qilinadi.

Neyron Tarmoqlarning tuzilishi va ishlashi.

Neyron tarmoqlar biologik asab tizimlarining sun'iy modeli sifatida yaratilgan bo'lib, ma'lumotlarni qayta ishlash, o'rganish va tanishish kabi masalalarda samarali ishlaydi. Neyron tarmoqlar quyidagi asosiy elementlardan tashkil topadi:

1. Kiritish qatlami (Input Layer): Ushbu qatlam neyron tarmoqqa tashqi ma'lumotlarni qabul qiladi. Bu ma'lumotlar ko'pincha raqamli shaklda bo'ladi va kiritish neyronlariga yuboriladi.

2. Yashirin qatlamlar (Hidden Layers): Kiritish qatlami orqali kelgan ma'lumotlar yashirin qatlamlarda qayta ishlanadi. Yashirin qatlamlar bir yoki bir nechta bo'lishi mumkin va ular neyron tarmoqning samaradorligiga ta'sir ko'rsatadi. Bu qatlamlarda ma'lumotlar transformatsiya qilinadi va yanada murakkab xususiyatlarga ega bo'ladi.

3. Chiqarish qatlami (Output Layer): Bu qatlamda neyron tarmoqqa kiritilgan ma'lumotlar asosida chiqarish amalga oshiriladi. Ya'ni, neyron tarmoq o'qitilgan model yordamida taxminiy natijani taqdim etadi.

O'rganish mexanizmi

Neyron tarmoqlar odatda backpropagation algoritmi yordamida o'rganadi. Bu algoritmi tarmoqning chiqishidagi xatolikni aniqlab, xatolikni minimallashtirish uchun og'irliklarni yangilaydi. Neyron tarmoqning har bir qatlamidagi neyronlar ma'lumotlarni qayta ishlashda o'z vaznlarini yangilash orqali yanada samaraliroq natijalarga erishadi.

Neyron tarmoqlarning amaliy dasturlari

Neyron tarmoqlar ko'plab sohalarda muvaffaqiyatli qo'llanilmoqda. Ularning ba'zi amaliyotlari quyidagilarni o'z ichiga oladi:

1. Kompyuter ko'rish (Computer Vision): Tasvir va videolarni analiz qilishda qo'llaniladi. Ob'ektlarni tanib olish va tasvirlarni tasniflashda juda samarali.

2. Tabiiy tilni qayta ishlash (NLP): Matnlarni tushunish, tarjima qilish va nutqni analiz qilish jarayonlarida qo'llaniladi.

3. Ovozli tanib olish: Ovoz signallarini qayta ishlash va nutqni yozuvga o'tkazishda foydalaniladi.

4. Rejalashtirish va optimallashtirish: Murakkab tizimlar va logistika muammolarini yechishda qo'llaniladi.

Neyron to'rlarning ishlash algoritmi pythonda misol:

```
import numpy as np
```

```

# Aktivatsiya funksiyasi (sigmoid)
def sigmoid(x):
    return 1 / (1 + np.exp(-x))
# Sigmoid funksiyasining hosilasi
def sigmoid_derivative(x):
    return x * (1 - x)
# Neyron to'rlar sinfi
class NeuralNetwork:
    def __init__(self, inputs, outputs):
        self.inputs = inputs
        self.outputs = outputs
        self.weights = np.random.rand(self.inputs.shape[1], self.outputs.shape[1])
        self.bias = np.random.rand(1, self.outputs.shape[1])
    # O'rganish jarayoni
    def train(self, iterations):
        for i in range(iterations):
            output = self.feedforward()
            error = self.outputs - output
            adjustments = error * sigmoid_derivative(output)
            self.weights += np.dot(self.inputs.T, adjustments)
            self.bias += np.sum(adjustments, axis=0, keepdims=True)
    # Feedforward jarayoni
    def feedforward(self):
        return sigmoid(np.dot(self.inputs, self.weights) + self.bias)
# Dastur misoli
inputs = np.array([[0, 0], [0, 1], [1, 0], [1, 1]])
outputs = np.array([[0], [1], [1], [0]])
nn = NeuralNetwork(inputs, outputs)
nn.train(10000)
print("Neyron to'r natijalari: \n", nn.feedforward())

```

Yuqoridagi kod oddiy neyron to'rda ikkita kirish, bitta yashirin qatlam va bitta chiqishdan foydalanib XOR operatsiyasini o'rganadi.

Genetik algoritmlarning tuzilishi

Genetik algoritmlar biologik evolyutsiyaga asoslangan algoritmlar bo'lib, ular tabiiy tanlanish va mutatsiya jarayonlarini modellashtiradi. Genetik algoritmlar quyidagi asosiy jarayonlar asosida ishlaydi:

1. Tanlash (Selection): Genetik algoritmda mavjud yechimlar orasidan eng samarali yoki eng mos keluvchi yechimlar tanlanadi. Bu tanlash jarayoni tabiiy tanlanishga o'xshaydi va har bir yechimning moslik darajasi (fitness) asosida amalga oshiriladi.

2. Chatishtirish (Crossover): Tanlangan yechimlar o‘zaro bir-biriga gen almashadi va yangi yechimlar hosil qiladi. Bu jarayon biroz evolyutsiyaga o‘xshash bo‘lib, yangi kombinatsiyalar yordamida yaxshiroq yechimlar topishga yordam beradi.

3. Mutatsiya (Mutation): Yangi yaratilgan yechimlarga kichik o‘zgarishlar kiritiladi. Bu o‘zgarishlar genlarning yaxshilanishiga olib kelishi va yechimlarni yanada optimallashtirishga yordam beradi.

Amaliy dasturlari

Genetik algoritmlar quyidagi sohalarda keng qo‘llaniladi:

- Tarmoq optimallashtirish: Ma’lumotlar uzatish tarmoqlari samaradorligini oshirish uchun.

- Sanoat jarayonlarini avtomatlashtirish: Ishlab chiqarish jarayonlarini optimallashtirishda qo‘llaniladi.

- Murakkab muhandislik masalalari: Texnologik muammolarni yechishda qo‘llaniladi.

- Sun’iy intellekt: Sun’iy intellekt tizimlarida optimallashtirish va yangi yechimlarni topishda foydalaniladi.

Neyron tarmoqlar va genetik algoritmlar integratsiyasi.

Neyron tarmoqlar va genetik algoritmlar o‘zaro birlashtirilganda, ko‘proq samaradorlikka erishiladi. Genetik algoritmlar neyron tarmoqlarning hiperparametrlarini optimallashtirishda ishlatiladi. Hiperparametrlar deganda qatlamlar soni, neyronlar soni va o‘qitish tezligi kabi parametrlar tushuniladi. Ushbu parametrlarni optimallashtirish, tarmoqning o‘rganish samaradorligini oshiradi va aniqlikni yaxshilaydi.

Bunday integratsiya, asosan, quyidagi sohalarda amaliyotlarda ishlatiladi:

- Tasvirlarni tasniflash: Rasm yoki videolarni aniqlash va kategoriyalarga ajratish.
- Prognozlash: Ma’lumotlar asosida kelajakdagi hodisalarni oldindan bashorat qilish.
- Tabiiy tilni qayta ishlash (NLP): Matnli ma’lumotlarni chuqur tahlil qilish va tushunishda samarali yechimlarni taklif qiladi.

Quyida Python dasturlash tilida genetik algoritmnining sodda bir misolini ko‘rib chiqamiz. Bu algoritim bizga eng yaxshi yechimni topish uchun ishlaydi. Masalan, biz bir xil funksiyani minimallashtirishga harakat qilamiz, masala: $f(x)=x^2$, bu yerda x ni topish kerak.

Genetik algoritim tuzilmasi:

1. Populyatsiya yaratish: Tasodifiy qiymatlardan iborat birinchi populyatsiyani yaratish.
2. Fitnes funksiyasi: Har bir individning qanchalik yaxshi ekanini o‘lchaydigan funksiya.

3. Tanlash: Eng yaxshi individlarni tanlash.

4. Chatishtirish: Ota-onalar o‘rtasida almashish amalga oshiriladi.

5. Mutatsiya: Genlar ozgina o‘zgaradi.

6. Yangi avlod: Yangi populyatsiya yaratiladi va algoritim davom etadi.

Python'da genetik algoritim misoli:

```
import random
```

```
# Parametrlar
```

```
POPULATION_SIZE = 10
MUTATION_RATE = 0.01
GENERATIONS = 100
# Fitnes funksiyasi (minimallashtirish maqsadi - masala:  $f(x) = x^2$ )
def fitness_function(x):
    return x ** 2
# Tasodifiy populyatsiya yaratish
def create_population(size, lower_bound, upper_bound):
    return [random.uniform(lower_bound, upper_bound) for _ in range(size)]
# Tanlash: eng yaxshi individlarni tanlaymiz (bu yerda elitizm qo'llaniladi)
def selection(population):
    return sorted(population, key=lambda x: fitness_function(x))[:2]
# Chatishtirish (Crossover): ikki individdan yangi avlod yaratish
def crossover(parent1, parent2):
    crossover_point = random.uniform(0, 1)
    child1 = crossover_point * parent1 + (1 - crossover_point) * parent2
    child2 = crossover_point * parent2 + (1 - crossover_point) * parent1
    return child1, child2
# Mutatsiya: kichik o'zgarishlar kiritish
def mutate(individual, mutation_rate):
    if random.random() < mutation_rate:
        return individual + random.uniform(-0.1, 0.1) # O'zgina o'zgarish
    return individual
# Genetik algoritm jarayoni
def genetic_algorithm():
    # Boshlang'ich populyatsiya
    population = create_population(POPULATION_SIZE, -10, 10)
    # Avlodlar bo'yicha iteratsiya
    for generation in range(GENERATIONS):
        # Eng yaxshi individlarni tanlash
        parents = selection(population)
        # Yangi populyatsiya yaratish
        new_population = []
        while len(new_population) < POPULATION_SIZE:
            parent1, parent2 = random.sample(parents, 2)
            child1, child2 = crossover(parent1, parent2)
            # Mutatsiya
            child1 = mutate(child1, MUTATION_RATE)
            child2 = mutate(child2, MUTATION_RATE)
            # Yangi avlod qo'shish
            new_population.append(child1)
```

```

new_population.append(child2)
population = new_population[:POPULATION_SIZE]
# Eng yaxshi individni chop etish
best_individual = min(population, key=lambda x: fitness_function(x))
print(f"Generation {generation + 1}: Best = {best_individual}, Fitness = {fitness_function(best_individual)}")
# Eng yaxshi yechimni qaytarish
return min(population, key=lambda x: fitness_function(x))
# Genetik algoritmnini ishga tushirish
best_solution = genetic_algorithm()
print(f"Eng yaxshi yechim: {best_solution}, Fitnes qiymati: {fitness_function(best_solution)}")

```

Algoritmning ishlash bosqichlari:

1. Dastlabki populyatsiya tasodifiy qiymatlarda yaratiladi.
2. Har bir individning fitnes qiymati hisoblanadi .
3. Eng yaxshi individlar tanlab olinadi va ular o‘zaro gen almashish orqali yangi avlodlar yaratadi.
4. Har bir avlodda kichik mutatsiyalar kiritilib, yangi populyatsiya yaratiladi.
5. Har bir avlodagi eng yaxshi yechim chiqadi va fitnes qiymati hisoblanadi.
6. 100 ta avloddan keyin eng yaxshi yechim ekranga chiqariladi.

Bu misol genetik algoritmnining asosiy komponentlarini ko‘rsatib beradi va optimallashtirish uchun ishlatiladi.

Xulosa: Neyron tarmoqlar va genetik algoritmlar zamonaviy kompyutertexnologiyalarining ajralmas qismiga aylandi. Ushbu texnologiyalar o‘zaro integratsiya qilinganda, murakkab muammolar samarali yechiladi. Neyron tarmoqlar o‘z-o‘zini o‘rganish qobiliyatiga ega bo‘lgan tizimlar yaratishda katta rol o‘ynaydi. Genetik algoritmlar esa evolyutsiyaga asoslangan yondashuvlar orqali neyron tarmoqlarni yanada optimallashtirishga xizmat qiladi. Bu ikki texnologiyaning birgalikdagi ishlashi texnologik rivojlanish jarayonini yanada kuchaytiradi va yangi bosqichga olib chiqadi.

FOYDALANILGAN ADABIYOTLAR:

1. Abdulaxadov, N., Saminjonov, S., & Tojimamatov, I. (2023). MA'LUMOTLAR VA AXBOROTLARNI VIZUALIZATSIYA QILISH USULLARI. ЕВРАЗИЙСКИЙ ЖУРНАЛ ТЕХНОЛОГИЙ И ИННОВАЦИЙ.
2. Tojimamatov, I. N., Mamalatipov, O., Rahmatjonov, M., & Farhodjonov, S. (2023). NEYRON TARMOQLAR. НАУКА И ИННОВАЦИЯ.
3. Tojimamatov, I. N., & Karimova, N. (2022). SUN'IIY NEYRON TARMOQLARINI O'QITISH USULLARI. ORIENTAL RENAISSANCE: INNOVATIVE, EDUCATIONAL, NATURAL AND SOCIAL SCIENCES.

4. Tojimamatov, I. N., Olimov, A. F., Khaydarova, O. T., & Tojiboyev, M. M. (2023). CREATING A DATA SCIENCE ROADMAP AND ANALYSIS. PEDAGOGICAL SCIENCES AND TEACHING METHODS, 2(23), 242-250.
5. Gulhayo, M., Gulnoza, A., & Isroil, T. (2023). MA'LUMOTLARNI QAYTA ISHLASHDA ERP TIZIMLARI. MA'LUMOTLARNI QAYTA ISHLASHDA SAP TIZIMLARI. Journal of Integrated Education and Research, 2(4), 87-89.
6. Isroil, T. (2023). NOSQL MA'LUMOTLAR BAZASI: TANQIDIY TAHLIL VA TAQQOSLASH. IJODKOR O'QITUVCHI, 3(28), 134-146.
7. Saidjamolova, B. M., & Tojimamatov, I. N. (2023). BIZNESDA «BIG DATA» TEXNOLOGIYALARI VA ULARNING AHAMIYATI. Лучшие интеллектуальные исследования, 11(4), 56-63.
8. Tojimamatov, I. N., Topvoldiyeva, H., Karimova, N., & Inomova, G. (2023). GRAFIK MA'LUMOTLAR BAZASI. Евразийский журнал технологий и инноваций, 1(4), 75-84.
9. Tojimamatov, I., & Rahmatov, Z. (2024). QATIYMAS NEYRON TARMOQLARI. Инновационные исследования в науке, 3(5), 118-124.
10. Nurmatovich, T. I. (2024). GENETIK ALGORITM VA UNING QO'LLANILADIGAN MASALALARI. worldly knowledge conferens, 8(1), 172-175.
11. Tojimamatov, I., & Madaminova, K. (2024). FORMAL GRAMMATIKA VA SEMANTIK TO'R. Development and innovations in science, 3(5), 53-61.
12. Онаркулов, М. К. (2024). ИНТЕГРАЦИЯ НЕЙРОННЫХ СЕТЕЙ В РАСПОЗНАВАНИИ РЕЧИ И РАЗРАБОТКА НОВЫХ МЕТОДОВ. ОБРАЗОВАНИЕ НАУКА И ИННОВАЦИОННЫЕ ИДЕИ В МИРЕ, 47(7), 193-197.
13. Онаркулов, М. К., угли Юсупов, М. А., & угли Умиржонов, Л. А. (2023). ПРИМИНЕНИЕ НЕЙРОННЫХ СЕТЕЙ В РАСПОЗНАВАНИИ РЕЧИ. Educational Research in Universal Sciences, 2(3), 1206-1210.
14. Umarjon o'g, Y. Y. L. (2024). PROBLEMS AND MODERN TRENDS IN COMPUTING ENGINEERING. Multidisciplinary and Multidimensional Journal, 3(6), 17-21.
15. Onarkulov, M., & Nabijonov, A. (2024). DB2: KATTA HAJMDAGI MA'LUMOTLARNI SAMARALI BOSHQARISH. Инновационные исследования в науке, 3(5), 99-104.
16. Onarkulov, M., & G'oyibova, G. (2024). SQL SERVER ILOVASIDA UNIVERSAL MODELNI QO'LLASH VA KONVERTATSIYA ETISHNING SAMARALI YO'LLARI. Академические исследования в современной науке, 3(18), 147-152.
17. Karimberdiyevich, M. O. (2024). RBF TURLARINING O 'QITISH ALGORITMI VA XOR MASALASI. worldly knowledge conferens, 8(1), 176-180.
18. Karimberdiyevich, M. O. (2024). GAUSS FUNKSIYASI. worldly knowledge conferens, 8(1), 239-244.

19. Karimberdiyevich, O. M. (2024). SQL TILIDA PROTSEDURA VA FUNKSIYALARDAN FOYDALANISHNING AHAMIYATI. worldly knowledge confere, 8(1), 145-148.

20. Karimberdiyevich, O. M. (2024). BIR QATLAMLI NEYRON TO‘RLARI VA ULARNI YARATISH USULLARI. IQRO INDEXING, 9(2), 104-108.

21. Onarkulov, M., & Omonaliyeva, E. (2024). QARORLAR DARAXTI VA UNI KIRITISH ALGORITIMI. Science and innovation in the education system, 3(6), 66-73.

22. Onarkulov, M., & Isaqova, S. (2024). NEYROCHILAR, MAXSUS MATRITSALI KUCHAYTIRGICHLAR VA NEYROEMULYATORLAR. Science and innovation in the education system, 3(6), 52-58.

23. Karimberdiyevich, M. O. (2024). EKSPERT TIZIMLARI YARATISH VA ULARNING MUAMMOLARI. ILM-FAN YANGILIKLARI KONFERENSIYASI, 2(1), 123-126.

24. Onarkulov, M., & Meliboyeva, A. (2024). HEMMING NEYRON TO‘RLAR VA ULARNING ARXITEKTURASI. Current approaches and new research in modern sciences, 3(4), 177-181.

25. Onarkulov, M., & Satinova, G. (2024). NEYRON TO‘RLARIDA FAOLLASHTIRISH FUNKSIYALARI. Theoretical aspects in the formation of pedagogical sciences, 3(8), 26-30.

26. Karimberdiyevich, O. M., & Mahamadamin o‘g‘li, Y. A. (2023). BASHORATLI TAHLILLAR UCHUN MASHINALI O‘QITISH ALGORITMLARI. QIYOSIY QARASHLAR. Journal of Integrated Education and Research, 2(1), 130-134.