

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

**«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИМЕНИ Н. Г. ЧЕРНЫШЕВСКОГО»**

Кафедра дискретной математики и информационных технологий

**РАЗРАБОТКА ПРИЛОЖЕНИЯ ДЛЯ ОПТИМИЗАЦИИ
НЕЙРОННЫХ СЕТЕЙ С ПРИМЕНЕНИЕМ ГЕНЕТИЧЕСКИХ
АЛГОРИТМОВ**

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

студента 4 курса 421 группы
направления 09.03.01 — Информатика и вычислительная техника
факультета КНиИТ
Крылова Романа Алексеевича

Научный руководитель

к. э. н., доцент

Г. Ю. Чернышова

Заведующий кафедрой

к. ф. - м. н., доцент

Л. Б. Тяпаев

Саратов 2024

ВВЕДЕНИЕ

Эксперименты по обучению нейронных сетей показали, что известные методы локальной и глобальной оптимизации (градиентные, стохастические, ньютоновские, гессианские и т.д.) требуют значительного количества шагов обучения, чувствительны к точности вычислений, требуют значительного количества дополнительных переменных, следовательно, актуальной задачей является поиск и разработка новых методов обучения нейронных сетей.

Целью работы является оптимизация обучения нейросетевых моделей с помощью генетического алгоритма.

К задачам бакалаврской работы относятся анализ возможностей генетических алгоритмов для обучения искусственных нейронных сетей, разработка web-приложения для обучения нейронной сети с помощью генетического алгоритма и оценка точности полученных моделей.

Объектом исследования данной работы является генетический алгоритм.

Предметом исследования является разработка инструментальных средств для обеспечения возможности применения генетического алгоритма.

Данная бакалаврская работа включает в себя три раздела. Первый раздел посвящен обзору технологии нейронных сетей и генетическому алгоритму. Второй раздел отведен под реализацию web-приложения для обучения нейросети на определенном наборе данных с помощью этого алгоритма. Третий раздел включает в себя результаты работы данного приложения и подведение статистики по прошедшему обучению.

Бакалаврская работа включает семнадцать рисунков, одну таблицу и двадцать используемых источников.

В качестве материалов в бакалаврской работе использовались научные издания, учебники, учебные пособия, документация библиотекам Pandas, PyGAD и Streamlit на платформе Python, выборки в виде наборов данных по классификации астероидов и отзывов авиакомпаний.

КРАТКОЕ СОДЕРЖАНИЕ РАБОТЫ

Первая глава

Нейронная сеть — это совокупность нейронов, определенным образом связанных друг с другом [1, 2]. Нейрон — это элемент, который вычисляет выходной сигнал из набора входных сигналов. То есть основная последовательность действий нейрона:

- прием сигналов от предыдущих элементов сети;
- комбинирование входных сигналов;
- вычисление выходного сигнала;
- передача выходного сигнала следующим элементам нейронной сети.

Архитектура нейронной сети ее описывает структуру и конфигурацию, включая количество слоев, количество нейронов на одном слое, функцию активации, метод оптимизации и другие параметры, которые определяют, как сеть обрабатывает входные данные и выдает результаты [3].

Прямые нейронные сети называются многослойными персептронами. Они состоят из входного слоя, нескольких скрытых слоев и выходного слоя. Каждый слой состоит из нейронов, которые обрабатывают входные данные и отправляют их на следующий уровень. Прямые нейронные сети обычно используются для задач классификации и регрессии [4].

В реализованном приложении было решено использовать многослойный персептрон из-за того, что он является гибким и эффективным способом решения многих задач классификации, и его несложно построить. В то время как традиционные линейные классификаторы могут не справиться с этой задачей, MLP известен своей способностью моделировать сложные нелинейные взаимосвязи в данных.

Существуют различные подходы к построению персептронных моделей основанных на методе градиентного спуска и его модификациях:

- пакетный градиентный спуск;
- стохастический градиентный спуск;
- минипакетный градиентный спуск [5].

Они могут быть использованы для нахождения экстремального значения целевой функции, но не всегда представляется возможным проверить, получено ли значение глобального экстремума. Нахождение локального экстремума вместо глобального называется преждевременной сходимостью. По-

мимо проблемы преждевременной сходимости, существует еще одна проблема — время процесса вычисления. Более точные методы оптимизации обычно требуют много времени для работы.

Основная идея генетического алгоритма заключается в организации борьбы за существование и естественного отбора среди пробных решений некоторой оптимизационной задачи. В генетическом алгоритме в основном применяются биологические аналогии, поэтому и терминология этого процесса носит биологический характер. Таким образом, одно пробное решение мы будем называть особью или хромосомой, а совокупность всех пробных решений — популяцией.

Приспособленность особей в условиях естественного отбора позволяет определить, какие особи будут участвовать в развитии колонии, а какие — нет. В данном случае приспособленность определяется целевой функцией [6].

Пороговая величина в селекции может быть вычислена разными способами. Поэтому в литературе по ГА выделяют различные вариации селекции. Наиболее известные из них — это турнирный и рулеточный (пропорциональный) отборы [7].

Первым шагом в алгоритме является синтез случайных параметрических моделей, которые являются индивидуальными с точки зрения генетических алгоритмов. Затем определяется значение функции приспособленности. Это наиболее трудоемкий шаг, поскольку на нем происходит обучение нейросетевых моделей для новых параметрических моделей. Затем происходит проверка, не было ли достигнуто требуемое значение функции пригодности.

Вторая глава

После рассмотрения различных вариантов в качестве языка разработки было решено выбрать Python. Это не единственный и не самый лучший вариант для разработки web-приложений, но он был выбран потому, что для него написано большое количество библиотек, у него есть открытый исходный код, а также на нем можно разрабатывать как front-часть приложения, так и back-часть.

После рассмотрения различных библиотек с функционалом обработки данных выбор был сделан в пользу пакета Pandas.

Pandas — программная библиотека написанная на языке Python для анализа и обработки данных. Работа с данными строится с помощью библио-

теки NumPy, которая является инструментом более низкого уровня. Данная библиотека предоставляет специальные структуры данных и операции для взаимодействия с числовыми таблицами и временными рядами.

Для анализа качества обучения нейронной сети была выбрана библиотека Scikit-learn.

Scikit-learn (ранее известная, как scikits.learn, а также известная как sklearn) — библиотека, предназначенная для машинного обучения, написанная на языке программирования Python и распространяемая в виде свободного программного обеспечения.

Одним из главных элементов разрабатываемого web-приложения является система обучения нейронной сети с помощью генетического алгоритма. Для решения этой задачи было решено выбрать библиотеку PyGAD.

PyGAD — является библиотекой Python с открытым исходным кодом для построения генетического алгоритма и оптимизации алгоритмов машинного обучения. Также у нее есть возможность работать с Keras и PyTorch [8].

Неотъемлемой частью любого web-приложения является графический интерфейс. В данной работе web-интерфейс реализован с помощью фреймворка Streamlit.

Streamlit — это фреймворк для языка программирования Python. Он содержит набор программных инструментов, которые помогают перенести модель машинного обучения в web. Написанную программу с помощью этого фреймворка можно быстро сделать web-приложением.

Основные возможности Streamlit:

1. быстрое развертывание;
2. использование скриптов;
3. виджеты и визуализация [9].

Перед разработкой приложения необходимо удостовериться в синхронизации всех используемых библиотек. Для этого было решено создать набор собственных классов.

В состав этого набора будут входить:

1. **Класс адаптер данных.** В данном классе будет реализовано хранение и редактирование набора данных.
2. **Класс адаптер генетического алгоритма.** Используется для создания и запуска генетического алгоритма, а также для сбора характери-

стических данных о результатах его работы.

3. Класс параметров. Необходим для хранения, проверки на ошибки и нормализации параметров, используемых генетическим алгоритмом.

Важным этапом процесса машинного обучения является предобработка данных. Удаление атрибутов, избавление от пустот и аномалий — все это важные критерии повышения скорости обучения и увеличения верхнего порога точности нейронной сети.

При инициализации данного класса в первую очередь происходит настройка библиотеки Pандас: разрешаются цепные назначения и при выводе набора данных в виде таблицы не происходит сокращения числа атрибутов. Далее создаются поле для хранения данных и его резервная копия на случай восстановления и поля для нецелевого и целевого атрибутов.

Чтобы достичь наибольшей точности в обучении нейронной сети, необходимо выполнить несколько действий по упрощению набора данных. Во-первых, избавиться от пропущенных данных — либо заменить пропуски на среднее значение, либо удалить их полностью. Во-вторых, если пропуски составляют большую часть атрибута, стоит удалить его полностью, так как оно будет только мешать процессу обучения.

Так как нейросети работают только с числами, возникает необходимость в кодировании категориальных атрибутов. Одним из алгоритмов кодирования является порядковое кодирование (ordinal encoder) — последовательная нумерация всех уникальных значений атрибута. Такой метод применяется, когда можно утверждать, что кодируемые значения имеют свойства порядковых данных. Для преобразования номинальных данных используется другой алгоритм — номинальное кодирование (one hot encoder) — каждое уникальное значение кодируемых данных преобразуется в новый атрибут, в котором значение единицы означает принадлежность записи к этому значению, а ноль — нет.

К некоторым другим методам относятся: метод стилизации набора данных для его отрисовки в web-части, полное восстановление данных и их нормализация, а также несколько методов для проверки корректности полученных от пользователя запросов.

Генетический алгоритм — метод оптимизации, возможности которого позволяют решать широкий спектр задач, в том числе и обучение нейрон-

ных сетей. В разрабатываемом приложении используется библиотечный алгоритм, однако, помимо него, так же необходимы и средства для оценки качества обучения сети. В дополнение к этому важно иметь функционал для отслеживания результативности промежуточных этапов алгоритма. По вышеперечисленным причинам было решено реализовать дополнительный класс, который будет удовлетворять всем этим требованиям.

Функционал данного класса поделен на три основных аспекта: инициализация шаблона нейронной сети и генетического алгоритма, отслеживание эффективности развития сети на каждом этапе и составление полной статистики на момент конца обучения.

Параметры генетического алгоритма представлены разными типами и каждый из них обладает собственными ограничениями, которые могут быть зависимы друг от друга. В разрабатываемом приложении не будут использоваться все характеристики, которые можно задать при инициализации класса, чтобы не усложнять интерфейс настройки для пользователя.

Данный класс исполняет функции проверки корректности выбранных пользователем настроек генетического алгоритма. По этой причине в своем составе он будет иметь преимущественно поля, сеттеры и геттеры.

Рассмотрим подробнее такие группы параметров, как числовые параметры, параметры, которые задаются символьными строками и другие, для которых требуется более специфический анализ.

В состав числовых параметров, необходимых генетическому алгоритму входит количество особей в колонии. Их должно быть положительное количество и, в виду ограничений производительности аппаратуры, верхний предел составляет 1000 колоний. Этого будет достаточно для эффективного обучения. Количество поколений также должно быть ограничено. За период жизни одного поколения происходит скрещивание нейронных сетей, их мутация и отбор лучших представителей. При слишком больших значениях время обучения нейронной сети будет значительно возрастать. Ограничения этого параметра такие же как и у количества колоний. Число родителей не может превышать число особей в колонии. С другой стороны они могут в принципе отсутствовать, и тогда колония будет развиваться только за счет мутации генов.

В состав параметров, которые задаются символьными строками входит способ выбора родителей. Для качественного улучшения особей необходимо стимулировать их развитие и как колонии, и как индивидуальных существ. Одним из эффективных способов это сделать является разведение наиболее приспособленных особей, за что и отвечает выбор родителей в генетическом алгоритме. Есть несколько методов такого выбора, например: случайный выбор, стационарный выбор, метод турнира и другие. Сам параметр ограничивается заранее заложенным списком этих алгоритмов, из которого пользователь выбирает нужный. После выбора родителей необходимо определить, каким образом они сформируют колонию нового поколения. Передача генома от родителей к потомкам можем привести как к положительному развитию особи, так и к отрицательному, однако как-то конкретно повлиять на этот процесс нельзя. Существует возможность проконтролировать только распределение генов от родителей к будущему поколению. Для этого используются определенные алгоритмы скрещивания, например: одноточечное пересечение, двухточечное пересечение, равномерное пересечение и другие. Решающим фактором продуктивного развития особей колонии является выбор правильного метода мутации. И выбор родителей, и выбор способа скрещивания используются только для отбора наиболее перспективных особей, но не влияют непосредственно на развитие следующего поколения. Для процесса эволюции необходимо появление отклонений в геномах. Такие способы мутации, как случайная, адаптивная, инверсионная и другие, позволяют изменить особь таким образом, что она может приобрести совершенно новые качества, как положительные, так и отрицательные.

Также существуют и некоторые другие параметры, для которых необходим более специфичный анализ: процент мутрирующих генов особи, критерии остановки и другие.

Третья глава

Первый эксперимент проведем на наборе данных о классификации астероидов [10]. Набор данных взят с сайта Kaggle — система организации конкурсов по исследованию данных, а также социальная сеть специалистов по обработке данных и машинному обучению.

Исследуемый набор данных включает в себя 40 атрибутов, в которые входят размеры астероидов, скорость их полета, градусные меры отклонения и другие, и состоит из 4688 записей.

Было решено никак не обрабатывать данные, помимо кодирования категориальных свойств и общей нормализации. Целевым атрибутом было решено выбрать логический атрибут “Hazardous”, который показывает, представляет ли данный астероид угрозу для Земли.

Определим параметры генетического алгоритма для обучения искусственной нейронной сети, которая будет предсказывать эти данные:

- количество особей в поколении: 150;
- количество поколений: 50;
- количество родителей: 10;
- переход лучшей особи в новое поколение: 3;
- переход родителей в новое поколение: 0;
- метод мутации: метод мутации подкачки;
- вероятность мутации гена: 0.2;
- способ выбора родителей для селекции: метод турнира;
- способ скрещивания: двухточечное пересечение;
- разрешить повторяющиеся гены;
- критерии принудительной остановки: точность $> 90.07\%$.

Точность полученной модели достаточно высока. Проверим насколько точной окажется обученная нейронная сеть с тестовыми данными.

Разница в данных полученных от тренировочной выборки и тестовой выборки практически незаметна, а точность нейронной сети на тестовой выборке достаточно высока, что говорит об успешном обучении. Сравним метрики, полученные при тестировании модели обученной генетическим алгоритмом, с метриками, полученными при тестировании модели обученной методом градиентного спуска.

Проведем еще один эксперимент на более большой выборке о классификации удовлетворенности пассажиров авиакомпании [11].

Исследуемый набор данных включает в себя 25 атрибутов, в которые входят пол пассажиров, возраст, как часто они летают, в каком классе они летят и другие, и состоит из 103905 записей.

Было решено уменьшить набор данных ввиду технических ограничений, которые не позволяют обрабатывать столько большие объемы информации, до 10000 записей. Целевым атрибутом для нейронной сети станет атрибут “Satisfaction”, который и отвечает за удовлетворенность пассажиров.

Определим параметры генетического алгоритма для обучения искусственной нейронной сети, которая будет предсказывать эти данные:

- количество особей в поколении: 50;
- количество поколений: 150;
- количество родителей: 8;
- переход лучшей особи в новое поколение: 3;
- переход родителей в новое поколение: 0;
- метод мутации: метод случайной мутации;
- Процент мутирующих генов: 20-35%;
- способ выбора родителей для селекции: метод турнира;
- способ скрещивания: двухточечное пересечение;
- критерии принудительной остановки: точность $> 85.33\%$.

Проведенные вычислительные эксперименты показывают, что результаты, выдаваемые искусственными нейронными сетями, которых обучил генетический алгоритм, имеют высокий уровень качества.

Полученные результаты обосновываются наличием слабой зависимости между целевым и остальными атрибутами выборки. Во время обучения генетический алгоритм при достаточном количестве данных может сам снизить влияние того или иного атрибута на предсказанный результат, что значительно повышает значения метрик.

ЗАКЛЮЧЕНИЕ

Эксперименты по обучению нейронных сетей показали, что известные методы локальной и глобальной оптимизации требуют значительного количества шагов обучения и точности вычислений.

Для повышения надежности решений, принимаемых на основе нейронной сети, необходимо исследовать альтернативные алгоритмы, например, генетической оптимизации.

В бакалаврской работе было разработано web-приложение для обучения нейронных сетей с помощью генетического алгоритма, обладающее следующим функционалом: предварительная обработка данных, обучение искусственной нейронной сети генетическим алгоритмом, оценка точности полученной модели с помощью различных метрик и сохранение полученной модели.

С помощью разработанного приложения были построены модели многослойного персептрона с оценкой точности, что подтверждает возможность применения генетических алгоритмов для решения подобных задач.

Рассмотренные вычислительные эксперименты позволяют сделать вывод, что в отдельных случаях применение генетического алгоритма целесообразно. К таким случаям следует отнести вопрос построения нейронных сетей большей точности, нежели были получены классическими методами.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1 Efficient multi-objective neural architecture search via lamarckian evolution / Elsken T., Metzen J. H., Hutter F. — Published in International Conference, 2018. 21 с.
- 2 Elsken T., Metzen J. H., Hutter F. The search for neural architecture: an overview. / T. Elsken // Journal of Machine Learning Research. 2019. arXiv: 1808.05377v2. 15 с.
- 3 The history of neural networks / Development II, Inc. // 260 Amity Road. Woodbridge, CT 06525, 2015. 2 с.
- 4 Liang J. Z. Evolutionary Neural Architecture Search for Deep Learning / J. Z. Liang // Dissertation, The university of Texas at Austin. 2018. 196 с.
- 5 Гасников А. В. Современные численные методы оптимизации. Метод универсального градиентного спуска. / А. В. Гасников // Учебное пособие. Изд. второе, испр., М.: издательство МЦНМО, 2021. 272 с.
- 6 Sivanandam S. N., Deepa S. N. Introduction to Genetic Algorithms. / S. N. Sivanandam // Springer-Verlag Berlin Heidelberg. 2008. 453 с.
- 7 A Study of Genetic Algorithm and Crossover Techniques. / IJCSMC. 2019. Vol. 8, № 3. С. 335-344.
- 8 PyGAD — Python Genetic Algorithm. Documentation: [Электронный ресурс] URL: <https://pygad.readthedocs.io/en/latest/index.html> (дата обращения: 20.01.2024)
- 9 Streamlit documentation. Documentation: [Электронный ресурс] URL: <https://streamlit-docs.netlify.app/> (дата обращения: 07.03.2024)
- 10 NASA: Asteroids Classification. Dataset: [Электронный ресурс] URL: <https://www.kaggle.com/datasets/lovishbansal123/nasa-asteroids-classification> (дата обращения: 02.04.2024)
- 11 Airline passenger satisfaction. Dataset: [Электронный ресурс] URL: <https://www.kaggle.com/code/fringinglife/airline-passenger-satisfaction-part-1/notebook> (дата обращения: 05.04.2024)