Муниципальное бюджетное общеобразовательное учреждение

«Средняя общеобразовательная школа №30»

**Влияние нейросетей на программирование**

Подготовил ученик 10 Б класса

Новенко Никита Денисович

Учитель по информатике:

Иванова Наталья Юрьевна

Подольск, 2023 год

# 

[Введение](#_30j0zll)2

[1. Искусственные нейронные сети 4](#_3znysh7)

[1.1. История искусственного интеллекта. 4](#_2et92p0)

[1.2. Принцип работы нейронных сетей, их сильные и слабые стороны 4](#_tyjcwt)

[2. Нейросети в программировании 6](#_3dy6vkm)

[2.1 Нейросети для программистов 6](#_1t3h5sf)

[2.2 Нейросети для обучения программированию 8](#_4d34og8)

[Заключение 13](#_2s8eyo1)

[Список использованной литературы:](#_17dp8vu)15

### Введение

В настоящее время создание программного обеспечения требует большого количества усилий, поскольку большинство этапов процесса по-прежнему выполняется человеком. Но можно ли это изменить? В лекции «Нейронные сети: maths & magic» Максим Щербаков утверждает, что да, это возможно. Нейронные сети могут стать автоматизированными помощниками, которые решат задачу, основываясь на заранее определенных критериях и шаблонах, и при этом работая быстрее, чем человек. Но какие именно задачи способны решать нейросети? Развитие машинного обучения продолжается стремительными темпами, и в будущем мы можем ожидать, что машины будут заменять человека в рутинных задачах. Например, нейронные сети могут помочь в составлении технического задания, в выборе наиболее подходящей методологии разработки программного обеспечения для конкретного проекта и в тестировании программного обеспечения. Это поможет ускорить все этапы разработки и снизит их стоимость.

**Цель:** цель проекта - исследовать и доказать практическую эффективность использования нейросетей в развитии программирования, предложить практические инструменты и методы для улучшения процесса обучения программированию.

**Задачи**

1. Исследовать применение нейросетей в различных областях программирования;
2. Изучить возможности интеграции нейросетей в среды разработки программного обеспечения.
3. Сравнить результаты работы программистов с использованием и без использования нейросетей.
4. Проанализировать преимущества и недостатки использования нейросетей в программировании.
5. Ответить на поставленный вопрос: «Может ли нейросеть заменить программистов?»

**Методы** моего исследования вопроса: Интернет ресурсы, изучение и анализ литературы, наблюдение.

### 

### Искусственные нейронные сети

Искусственный интеллект - это эмуляция деятельности человеческого мышления, осуществляемая с помощью компьютера. Один из способов тестирования искусственного интеллекта - это тест Тьюринга, который заключается в диалоге, где человек должен определить, кто из собеседников - человек, а кто - машина. Однако, данный тест считается недостаточно точным, и в настоящее время применяются другие методы тестирования. Тем не менее, полная симуляция человеческого мышления компьютером кажется невозможным, а искусственные нейронные сети используются как помощник в некоторых конкретных областях.

### История искусственного интеллекта.

В 1956 году Джон МакКарти, создатель функционального программирования, впервые упомянул искусственный интеллект на конференции в Университете Дартмута. Ранее, в 1935 году Алан Тьюринг сформулировал идею такой системы и описал несуществующую вычислительную машину с безграничной памятью и сканером, сканирующим эту память. В 1950 году он спрогнозировал, что интеллектуальной можно будет считать те системы, которые в общении неотличимы от человека, и разработал специальный тест на эту тему. Кристофер Стрейчи создал первую успешную программу искусственного интеллекта в 1951 году, которая уже в 1952 году играла в шашки с человеком. В 1965 году Джозеф Вейценбаум создал программу Элиза, которая является прообразом современных голосовых помощников, а в 1973 году появилась первая беспилотная машина – Стэнфордская тележка. В середине 1990-х искусственный интеллект получил новое развитие, примером чего является суперкомпьютер IBM Deep Blue, который в 1997 году одержал победу в шахматной партии над чемпионом мира Гарри Каспаровым. Сегодня машины очень быстро обучаются и анализируют информацию, приобретая способности, которые ранее считались возможными только для человека.

### Принцип работы нейронных сетей, их сильные и слабые стороны

Нейронные сети — это тип компьютерной программы, которая может обучаться и делать прогнозы на основе входных данных. Они вдохновлены структурой и функциями человеческого мозга.

Проще говоря, нейронные сети состоят из множества взаимосвязанных узлов или «нейронов», обрабатывающих информацию. Каждый нейрон принимает один или несколько входных данных, применяет к ним математическую функцию и выдает результат. Затем этот вывод передается другим нейронам в сети.

Принцип работы нейросетей основан на имитации работы человеческого мозга. Она состоит из многих связанных нейронов, которые обрабатывают информацию и передают ее дальше. Каждый нейрон получает входные данные, складывает их и вычисляет результат, который передается следующим нейронам. Во время обучения сеть принимает множество входных данных и результатов, и корректирует веса связей между нейронами для повышения точности работы. Когда сеть обучена, она может предсказывать результаты на основе новых входных данных. Это позволяет использовать нейросети для решения различных задач, таких как распознавание изображений, анализ текстов или прогнозирование будущих событий.

Нейросети теперь способны создавать музыку, анализируя шедевры классической музыки. При этом отличить творение машины от музыки, созданной человеком, может быть практически невозможно без отличного понимания работы обучения машин. На примере большой комнаты с листочками, на которых написаны символы, нейросеть работает по алгоритму перекладывания листочков без понимания их содержания. Нейросеть не создает произведение, а просто следует определенным правилам при сочинении музыки без понимания процесса. Ответить на вопрос о том, могут ли машины создавать, является сложной задачей.

Машинное обучение - инструмент, способный решать множество задач, но как оно работает? Секрет заключается в обучении нейронных сетей. Точно так же, как в комнате с листочками, нейросеть может быть обучена алгоритму, который указывает, куда и как помещать листочки. Однако также есть вариант, где мы напрямую не указываем алгоритм, а вместо этого используем обучение с подкреплением. "Учитель" поощряет правильные действия нейросети, и она запоминает зависимости, которые могут быть использованы в будущем, когда ей необходимо принимать решения самостоятельно. Кроме того, существуют различные методы обучения, такие как с учителем, без учителя и с частичным привлечением учителя, которые применяются в зависимости от условий обучения нейронной сети.

Чтобы обучить нейросеть, используя метод обучения с учителем, нужно предоставить набор данных и соответствующие правильные ответы. После этого нейросеть может определить закономерности и дать правильные ответы на тестовом наборе данных. Обычно нейросеть чаще ошибается на тестовом наборе из-за переобучения, когда она адаптирована только к обучающим данным и не может обработать новые данные.

Обучение без учителя также требует подачи набора данных, но в этом случае неизвестны правильные ответы. Задача состоит в том, чтобы найти зависимости в данных. Обучение с частичным привлечением учителя используется для обучения нейросетей, способных обрабатывать звук. В этом случае нет размеченных данных, которые нужно ручным образом обработать.

Самым интересным методом является обучение с подкреплением, который позволяет нейросети учиться на примерах правильного и неправильного поведения. Примером успеха такого метода было участие нейросети OpenAI в чемпионате по игре Dota 2, где нейросеть безошибочно побеждала профессиональных игроков. Однако, нейросеть все еще может быть обманутой, и профессиональным игрокам удалось одержать победу благодаря обману.

### 2. Нейросети в программировании

Нейросети могут быть использованы в программировании, они способны существенно упростить работу программистов, решая за них простые задачи, предлагая способы оптимизации их кода, но также, нейросети способны обучать людей программированию, они способны объяснять им функции языков программирования, объяснять ход решения задач в программировании, предлагать своё решение задачи

### 2.1 Нейросети для программистов

Нейросети имеют огромное влияние на программирование, представляя собой новый способ автоматизации и оптимизации работы программиста. С их помощью можно решать разнообразные задачи, в том числе:

Анализ данных - нейросети могут обрабатывать большие объемы данных и выявлять в них закономерности и тренды.

Предсказание - с помощью нейросетей можно строить модели, которые будут делать прогнозы на основе данных.

Классификация - нейросети могут различать объекты по определенным признакам.

Оптимизация - с помощью нейросетей можно находить оптимальные решения для сложных задач.

Нейросети можно использовать для программирования в различных областях. Например, в Machine Learning и AI, обработке естественного языка, компьютерном зрении, мобильной разработке и др. С их помощью можно значительно ускорить процесс разработки, сократить время на написание кода и минимизировать ошибки.

Нейросети существенно упрощают программирование, убрав много рутинной работы, связанной с написанием кода. Они могут автоматически генерировать код и исполнять множество повторяемых задач, отбрасывая излишние детали и делая процесс более быстрым и эффективным.

Однако, несмотря на все преимущества, нейросети пока не могут полностью заменить программистов, как минимум в ближайшие десятилетия. Это связано с тем, что они могут решать только заранее заданные задачи, а человеческий мозг гораздо более гибок и способен работать в сложных и нестандартных ситуациях. Кроме того, нейросети нуждаются в постоянном обновлении и обучении, а также требуют специалиста, который бы мог принимать решения в случае ошибок.

Нейронные сети на текущий момент развития не могут полностью заменить программистов, поскольку они не способны полностью заменить интуицию и творческий подход, которые характерны для человека.

Нейронные сети вызываются к решению определенных задач и работают по заранее заданным правилам и алгоритмам. Они не способны сформулировать новые задачи или самостоятельно изменить свой входной алгоритм.

Однако нейронные сети могут существенно облегчить жизнь программистам, автоматизируя часть их работы. Например, обученные нейронные сети могут анализировать большие объемы данных и находить в них закономерности, что может помочь программистам разрабатывать более эффективные алгоритмы.

В целом, нейронные сети позволяют программистам работать быстрее и качественнее, не утомляясь рутинными напряженными операциями в разработке сложных систем. Нейронные сети могут быть обучены распознавать кодовые ошибки в программах и предлагать исправления. Некоторые нейронные сети могут генерировать код на основе заданных параметров или анализа существующего кода. Нейронные сети могут использоваться для тестирования программ и автоматического обнаружения ошибок и уязвимостей. Нейронные сети могут помочь в оптимизации производительности программы, анализируя ее работу и предлагая улучшения. Нейронные сети могут использоваться для автоматического создания документации на основе кода программы. В целом, нейронные сети могут существенно ускорить разработку программного обеспечения, а также сделать ее более точной и эффективной.

### 2.2 Нейросети для обучения программированию

Нейросети могут быть использованы для обучения людей программированию, выявления их ошибок, предложения решения задач и объяснения. Например, данная технология может применяться для:

1. Оценки навыков программирования – приложения и сервисы могут использовать нейросети, чтобы анализировать и оценивать навыки людей в программировании. На основе этой оценки могут быть предложены индивидуальные курсы обучения и упражнения для улучшения навыков.
2. Выявления ошибок – обученная нейросеть может анализировать код и выявлять потенциальные ошибки, предупреждая программиста о них. Это позволит существенно сократить время на поиск и устранение ошибок.
3. Предложения решения задач – нейросеть может использоваться для анализа задачи и предложения наиболее эффективного решения. В качестве примера, может быть использовано решение задач на математику, физику, алгоритмы и другие предметы.
4. Объяснения – нейросеть может создавать объясняющие системы, которые показывают программистам, как работает код, а также обучают, как решать определенные типы задач.

В целом, использование нейросетей для обучения программированию может значительно улучшить процесс обучения и повысить качество конечного продукта, а также сокращает время, затрачиваемое на поиск ошибок и решение различных задач.

Пример запроса для нейронной сети, требующий решить задачу. Задача взята из примеров заданий ЕГЭ по информатике, задание 23(см. рисунок 1):



Ответ нейросети (см. рисунок 2) на запрос (см. рисунок 1):

В своём ответе нейросеть предложила использовать рекурсию для решения задачи, при этом она ещё объяснила ход программы. Да, это простая программа для опытных программистов, но нейросеть её напишет куда быстрее, в считанные секунды, что в рамках большого объёма работу существенно сэкономит времени и сил программисту. Также, этот ответ можно вполне использовать для объяснения решения такого типа задач студентам, изучающим программирование, или подготавливающимся к ЕГЭ, что не только сэкономит время и силы учителей, но и даст огромные возможности к самообучению у студентов.

Нейросети также можно использовать для выявления собственных ошибок в коде(см рис. 3, см рис. 4)



В запросе нейросети необходимо указать как программа должна работать, попросить исправить ошибки в программе, нейросеть сама найдёт эту ошибку, предложит новое решение, но которое будет при этом опираться на старое, кардинально его не изменив, только если ошибка не сопровождает весь код.



По введённому запросу (см рис. 3) нейросеть выведет подобный результат, он содержит объяснение ошибки, объясняет что именно в вашем коде препятствует нормальной работе программы, предлагает код, с уже исправленной ошибкой. Плюс объяснения нейросети перед объяснением человека, что человеку может потребоваться полное изменение кода, чтобы он заработал, ведь далеко не все люди могут понять ход решения отдельного человека, нейросеть как бы работает с тем, что имеет, только в крайних случаях может предложить совершенно иное решение кода от вами предложенного, нейросеть также всегда объясняет свои действия понятным языком. Но не стоит исключать, что нейросеть - это база знаний человека, она не умеет мыслить, а значить некоторые задачи она не может решить, а именно те, где необходимо изобрести решение, а не действовать по заложенному алгоритму.

Пример использования нейросети для объяснения материала, а именно что может предложить нейросеть на подобный запрос: “Опиши понятным языком создание функции в пайтон, приведи примеры” (см. рис. 5)



В своём ответе (см. рис 5), нейросеть подробно объясняет возможность создания функций в пайтон, с приведением примеров.

### Заключение

Нейронные сети становятся все более популярными в программировании из-за их способности изучать шаблоны из данных без явного программирования. Вот некоторые плюсы и минусы использования нейронных сетей для программирования:

Плюсы:

1. Нейронные сети могут изучать сложные шаблоны из данных, которые было бы трудно закодировать в явном виде.

2. Они могут обрабатывать широкий спектр типов входных данных, таких как изображения, речь и текст.

3. Нейронные сети могут адаптироваться и улучшать свою производительность с течением времени, поскольку в них поступает больше данных.

4. Их можно использовать для различных задач, включая распознавание изображений, распознавание речи и обработку естественного языка.

5. Они могут автоматизировать задачи, которые в противном случае потребовали бы значительного ручного труда.

Минусы:

1. Нейронным сетям требуется большой объем данных для эффективного изучения закономерностей, получение которых может быть трудным или дорогим.

2. Они могут быть дорогостоящими в вычислительном отношении, требуя значительной вычислительной мощности и времени для обучения и запуска.

3. Нейронные сети трудно интерпретировать, что затрудняет понимание того, как они принимают решения.

4. Они могут страдать от переобучения, когда они слишком хорошо усваивают тренировочные данные и становятся менее эффективными при представлении новых данных.

5. Нейронные сети могут быть чувствительны к качеству и количеству входных данных, что может повлиять на их точность и производительность.

### Список использованной литературы:

* OpenAI - исследовательская лаборатория искуственного интеллекта (ссылка: https://openai.com/) - открытый доступ. 15.03.2023
* Хабр - сайт в формате системы тематических коллективных блогов с элементами новостного сайта (ссылка: <https://habr.com/ru/companies/sciberia/articles/499672/>) - открытый доступ. 16.03.2023
* Хабр - сайт в формате системы тематических коллективных блоговс элементами новостного сайта (ссылка: <https://habr.com/ru/companies/sciberia/articles/499672/>) - открытый доступ. 17.03.2023
* Лекция: «Нейронные сети: maths & magic» Максим Щербаков