Полоцкий государственный университет имени Евфросинии Полоцкой

Грибанов А.А., Чиникайло А.П.

**Создание искусственного интеллекта, способного обучаться прохождению уровней в компьютерной игре «geometry dash»**

Полоцк, 2024

Введение.

В мире современных технологий искусственный интеллект (ИИ) становится неотъемлемой частью нашей повседневной жизни. Он проникает в различные сферы, начиная от медицины и финансов, и заканчивая развлекательной индустрией. Инновации в области искусственного интеллекта не только улучшают эффективность наших действий, но и открывают новые возможности для творчества и развития.

Искусственный интеллект имеет уникальную способность обучаться и адаптироваться к различным задачам, превосходя человеческие способности во многих областях. Он способен анализировать большие объемы данных, выявлять закономерности и принимать решения на основе полученной информации. Этот подход делает искусственный интеллект незаменимым инструментом в решении сложных задач, требующих высокой точности и скорости реакции.

Одним из увлекательных примеров применения искусственного интеллекта является его способность обучаться прохождению уровней в компьютерной игре "Geometry Dash". Эта игра представляет собой захватывающее платформерное приключение, где игрок управляет кубом, преодолевая различные препятствия и ловушки на пути к победе. Создание искусственного интеллекта, способного успешно завершать уровни этой игры, представляет собой интересную задачу, требующую комбинации навыков в области компьютерного зрения, обработки данных и принятия решений в реальном времени.

В данной работе будет рассмотенны методы и подходы к созданию такого искусственного интеллекта, а также его потенциальные применения в других областях. Мы также проанализируем достижения и вызовы, стоящие перед исследователями и разработчиками в этой увлекательной области искусственного интеллекта.

Подход.

Для создания искусственного интеллекта, способного проходить уровни в игре "Geometry Dash", был использован упрощенный вариант генетического алгоритма. Это адаптивные эвристические алгоритмы поиска, которые относятся к большей части эволюционных алгоритмов. Генетические алгоритмы основаны на идеях естественного отбора и генетики. Это интеллектуальное использование случайного поиска, снабженного историческими данными, для направления поиска в область более высокой производительности в пространстве решений. Они обычно используются для создания высококачественных решений задач оптимизации и поиска.

После процесса отбора создание ребенка происходит на этапе воспроизводства. На этом этапе генетический алгоритм использует два оператора вариации, которые применяются к родительской популяции. Ниже приведены два оператора, участвующие в фазе воспроизведения:

* Кроссовер. Кроссовер играет наиболее важную роль на этапе воспроизводства генетического алгоритма. В этом процессе точка кроссовера выбирается случайным образом внутри генов. Затем оператор скрещивания меняет генетическую информацию двух родителей из текущего поколения, чтобы создать новую особь, представляющую потомство.

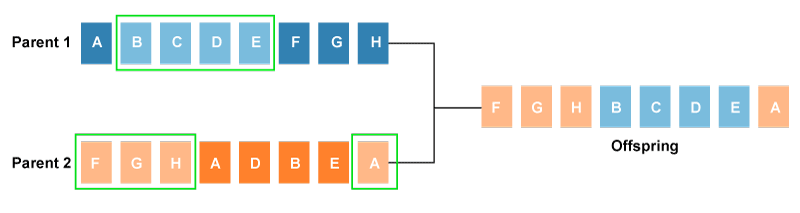


Рисунок 1 – пример кроссовера

* Мутация. Оператор мутации вставляет случайные гены в потомство (новый ребенок), чтобы поддерживать разнообразие популяции. Это можно сделать, перевернув некоторые биты хромосом. Мутация помогает решить проблему преждевременной конвергенции и усиливает диверсификацию.

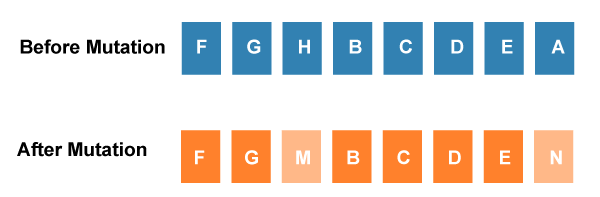


Рисунок 2 – Мутация

После фазы воспроизведения в качестве основания для прекращения применяется критерий остановки. Алгоритм завершает работу после достижения порогового решения пригодности. Он определит окончательное решение как лучшее решение в популяции.

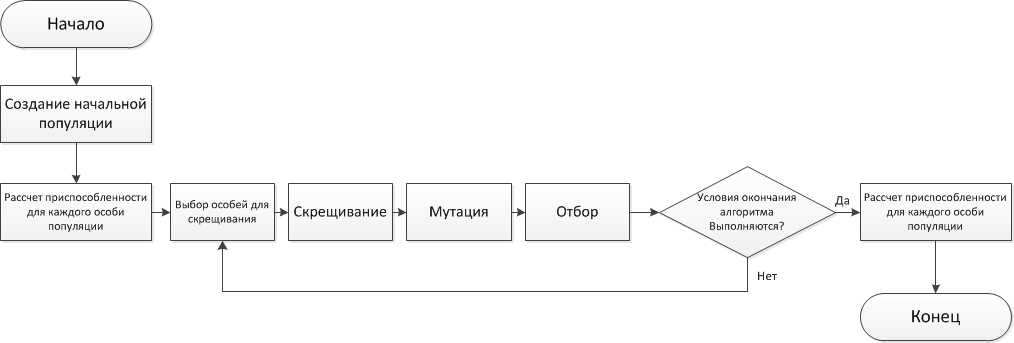


Рисунок 3 - Общий рабочий процесс простого генетического алгоритма

Это позволяет ускорить обучение в целом, но в случае, если ИИ застревает где-то, особенно близко к концу уровня, обучение может затянуться.

Принцип работы ИИ включает следующие этапы: сначала ИИ следует за списком сохраненных и подтвержденных как работающих прыжков. После того как все прыжки выполнены, он начинает случайно кликать. Если новые клики помогают ему продвигаться дальше по уровню, они сохраняются и используются для обновления списка прыжков.

Важно отметить, что экран не используется в качестве входных данных вообще - ИИ играет "вслепую". Единственные входные данные, которые он использует, - это текущая позиция X, режим игрока и завершен ли уровень.

Если ИИ не может продвинуться дальше после 10 попыток случайных прыжков, он удалит последний прыжок в списке. Чтобы обеспечить возможность нескольких откатов прыжков, новые прыжки сохраняются только если ИИ проходит точку, в которой он застрял. Если он все еще не может продвинуться дальше, он продолжает откатываться, пока не сможет продвинуться. Именно это позволяет ИИ учиться и самокорректироваться.

Для различных секций игры, таких как куб, шар, UFO и паук, хранящиеся клики сообщают программе, чтобы она кликала вниз и вверх. Для секций с кораблем, волной и роботом, хранящиеся клики говорят программе переключить мышь, позволяя ИИ удерживать кнопку мыши внизу. Это объясняет, почему аргумент строки режима необходим для правильной работы ИИ.

Следует отметить, что DashBot использует функции для чтения памяти процесса Geometry Dash. Некоторые антивирусные программы могут реагировать на использование этих функций как на вредоносное программное обеспечение.