

Розробка гри «Реверсі» з самонавчанням

Ян Юрійович Хамула

Кафедра інформатики та прикладної математики, Криворізький державний педагогічний університет, пр. Гагаріна, 54, м. Кривий Ріг, 50086, Україна
phlor@mail.ua

Анотація. *Метою дослідження є проектування та реалізація самонавчання для логічної гри «Реверсі». Задачами дослідження є аналіз наявних методів машинного навчання в логічних іграх; розробка алгоритму самонавчання. Об'єктом дослідження є машинне навчання, його розробка та функціонування. Предметом дослідження є розробка алгоритму вибору найкращого ходу в логічних іграх, який здійснюється завдяки поглибленому аналізу статистичних даних. Результат дослідження:* програмний продукт, що включає гру «Реверсі» та здатний до самонавчання алгоритм гри у неї.

Ключові слова: логічні ігри; машинне навчання; штучна нейрона мережа; реверсі; HTML5; JavaScript; NodeJS.

J. Yu. Khamula. Development of the “Reversi” game with self-learning

Abstract. *The objective of the study is the design and implementation of self-learning for logical game “Reversi”. The tasks of the study are to analyze the existing machine learning methods in logic games and to develop the algorithm of self-learning. The objects of the study are machine learning, its development and operation processes. The subject is the development of the algorithm that realizes choice of the best turn in logic games, which is carried through in-depth analysis of the statistics. The result is the software that contains the game “Reversi” and the self-learning algorithm of playing in it.*

Keywords: logical games; machine learning; artificial neural network; Reversi; HTML5; JavaScript; NodeJS.

E-mail: phlor@mail.ua.

У наш час комп'ютерні технології розвиваються дуже стрімко, зокрема, це відображається у різноманітті комп'ютерних ігор. Зазвичай комп'ютерні ігри вимагають від гравця великої кількості вмінь та навичок, при цьому самі вони є складними результатами роботи розробників програмного забезпечення та демонструють вміння та навички останніх. Найчастіше такі ігри дуже вимогливі до обчислювальних ресурсів, але при цьому не демонструють власного

інтелекту.

Разом з цим є досить багато логічних ігор, таких як шахи, шашки, го, хрестики-нулики, реверсі та інші. Програмування таких ігор вимагає від програми демонстрації власного інтелекту. Програма сама повинна приймати рішення і обирати продовження гри.

Тому, проаналізувавши наявні підходи до машинного навчання, ми вирішили розробити алгоритм на базі збору статистичних даних зіграних партій у базу даних для подальшого опрацювання. Спираючись на ці дані програма повинна приймати рішення, керуючись штучної нейронною мережею.

Проект розподілений на дві частини: клієнтську та серверну. Такий підхід зумовлений необхідністю розподілення навантаження обчислень під час роботи з базою даних.

Клієнтська частина відповідає за взаємодії з користувачем (інтерфейс) та розрахунок найкращого ходу з точки зору його пошуку методом Alpha-Beta відсічення на задану глибину.

Серверна – реалізує взаємодію з базою даних та приймає рішення стосовно вибору ходу. Поточна позиція та знайдений для комп'ютера хід передається клієнтом серверу. Після чого відбувається його підтвердження штучною нейронною мережею (ШНМ). На вхід ШНМ подається статистика результатів партій, що є в базі, з поточної позиції та на виході отримується підтвердження або відхилення знайденого ходу. У випадку відхилення, виконується пошук ходу на більшу глибину розрахунків. Підтвердивши даний варіант, сервер передає клієнту хід для здійснення.

Кожна подія, що відбувається під час розіграшу партії, зберігається в загальну базу як матеріал для подальшого навчання.

Підключена до сервера база даних має складну деревоподібну структуру. Вона містить ігрові позиції, інформацію про дії гравця, список ходів, зроблених на кожному етапі, і результат отриманий в партіях, що були зіграні раніше.

У спроектованому та реалізованому алгоритмі для самонавчання в логічній грі «Реверсі» навчання може здійснюватись на основі власних зіграних партій, або на основі імпортованої статистики інших ігор. Клієнтська частина розроблена як веб-застосунок. Графіка гри та інтерфейсу відображається за допомогою елементу HTML5 Canvas, алгоритм пошуку можливих ходів методом Alpha-Beta відсічення та інший функціонал реалізовано мовою JavaScript. Серверна частина гри розроблена засобами NodeJS. Зв'язок між ними здійснюється завдяки бібліотеці «socket.io».

Список використаних джерел

1. Корнилов Е. Н. Программирование шахмат и других логических игр / Е. Н. Корнилов. – СПб. : БХВ-Петербург, 2005. – 272 с.
2. Рассел С. Искусственный интеллект. Современный подход / Стюарт Рассел, Питер Норvig. – М. : Вильямс, 2007. – 1410 с.

References (translated and transliterated)

1. Kornilov E. N. Programmirovaniye shakhmat i drugikh logicheskikh igr [Programming of chess and other logical games] / E. N. Kornilov. – SPb. : BKhV-Peterburg, 2005. – 272 s. (In Russian)
2. Rassel S. Iskusstvennyi intellekt. Sovremennyi podkhod [Artificial Intelligence. The modern approach] / Stiuart Rassel, Piter Norvig. – M. : Viliams, 2007. – 1410 s. (In Russian)