

CZĘŚĆ I

Gry wideo i e-sport

Rozdział 1.

Sztuczna inteligencja i gry

1. Wprowadzenie

Budowa automatów i programów mogących konkurować z wybitnymi graczami fascynowała konstruktorów i naukowców od dawna. Skonstruowanie komputerów zintensyfikowało te prace. Od połowy lat pięćdziesiątych XX w., gdy pojawiła się koncepcja i nazwa sztucznej inteligencji, wiele jej ważnych kierunków badawczych, takich jak reprezentacja wiedzy, przeszukiwanie i wyszukiwanie, uczenie maszynowe, rozpoznawanie i rozumienie języka naturalnego, robotyka i systemy agentowe, wykorzystywało gry do testowania nowych koncepcji. Często nowe algorytmy i strategie sztucznej inteligencji budowane były na potrzeby doskonalenia programu dla gier. Gry stanowiły doskonałe środowisko do badań i testów. Bardzo ważnym czynnikiem, stymulującym naukowców i programistów była możliwość skonfrontowania osiągnięć z najlepszymi graczami, a więc wybitną ludzką inteligencją¹.

2. Najstarsze strategiczne gry planszowe – przeszukiwanie heurystyczne

W kontekście rywalizacji z inteligencją człowieka, szczególnie interesująca jest analiza postępu dla takich gier jak szachy, warcaby i go. Są to gry spełniające następujące kryteria: są skończone (liczba stanów oraz liczba ruchów dla każdego stanu jest skończona, w rozgrywce nie występują żadne elementy losowe, przebieg rozgrywki zależy wyłącznie od decyzji graczy); są z pełną

¹ X. Boming, Y. Xiaozhen, Adnan O.M Abuassba Recent Research on AI in Games, 2020 International Wireless Communications and Mobile Computing (IWCMC), Limasol 2020.

informacją (wszyscy gracze dysponują pełną informacją na temat stanu gry i przebiegu rozgrywki); są grami dwuosobowymi z sumą zerową (zwycięstwo jednego gracza oznacza porażkę drugiego). Programy grające w gry tego typu mają za zadanie, dla ustalonego stanu gry, wybrać „najlepszy” ruch. Najczęściej w tym celu budowane jest drzewo gry, w którym korzeniem jest aktualny stan gry, gałęziami są posunięcia możliwe do wykonania w danym położeniu, a węzłami są stany gry, do których prowadzą poszczególne ruchy. Liśćmi w drzewie są stany, w których gra dobiegła końca, tzn. zakończyła się zwycięstwem jednego z graczy lub remisem. Drzewo gry ma w przybliżeniu rozmiar wykładniczy względem średniej długości rozgrywki z danego stanu, przy czym podstawa jest tutaj związana ze średnią liczbą możliwych ruchów.

Liczba kombinacji szachowych jest nazywana liczbą Shannona i wynosi 10^{120} . To oszacowanie podał amerykański matematyk i inżynier *Claude Shannon* (uznawany za jednego z twórców teorii informacji). Liczba Shannona przedstawia wyliczoną z niedomiarem, liczbę gałęzi szachowego drzewa gry, co odpowiada 10^{120} możliwym partiom szachowym, które były, będą lub teoretycznie mogłyby być rozegrane. Kalkulacja Shannona opiera się na założeniu, że wykonanie pełnego ruchu w trakcie partii szachów, a więc dwóch półruchów wykonanych kolejno przez białe i czarne, daje średnio 1000 możliwych wariantów, zaś przeciętna partia szachów rozstrzyga się w ramach 40 ruchów. Liczba 10^{120} jest większa niż liczba atomów w obserwowalnym świecie. *C. Shannon* konkludował, że automat nie jest w stanie przeanalizować każdego możliwego ruchu, ponieważ obliczając jedno posunięcie w mikrosekundę obliczenia, musiałby zakończyć po końcu wszechświata. Jednakże jeżeli w szachach ograniczymy posunięcia tylko do tych racjonalnych – czyli takich, w których dwóch zawodników chce wygrać, liczba kombinacji wynosiłaby około 10^{48} . Niemniej w dalszym ciągu obracamy się w olbrzymiej przestrzeni możliwych rozwiązań. Gracz, który rozpatruje 8 ruchów w przód, wyobraża sobie tyle gier, ile jest gwiazd w galaktyce.

Dla gry w go liczba możliwych kombinacji wynosi aż 10^{761} . Dla gry w warcaby liczba racjonalnych kombinacji to już tylko około 10^{12} . Oczywiście jest więc konieczność redukcji rozpatrywanego drzewa gry. Pojawiają się liście, w których rozgrywka nie dobiegła końca i nieznanym jest wynik gry. Ten problem rozwiązuje się, gdy rozszerzymy ocenę stanu gry z trójwartościowej (zwycięstwo pierwszego gracza, remis, zwycięstwo drugiego gracza) do wartości liczbowej, określającej układ sił na planszy, szacującej szanse gracza na końcowe zwycięstwo. Do obliczania takiej wartości używa się funkcji oceniającej (nazywanej również funkcją ewaluacji). Do końca lat dziewięćdziesiątych XX w. postęp strategii i programów dla gier związany był z rozwojem metod inteligentnego przeszukiwania i doskonaleniem metod (również wykorzystujących heurystyki) obliczania i wartościowania stanów gry przy pomocy funkcji oceniających.

Historia budowy automatów i programów szachowych splata się z dokonaniem w automatyce i informatyce. Pierwszy prawdziwy automat szachowy² był zbudowany przez hiszpańskiego inżyniera *Torresa y Quevedo* w 1890 r. (zademonstrowany na wystawie światowej w 1914 r.). Potrafił on rozgrywać końcówkę król i wieża przeciwko królowi. W 1945 r. *Alan Turing* wykorzystał szachy do demonstracji możliwości komputera, a w 1950 r. zaprezentował pierwszy program szachowy (był on słabszy od przeciętnych graczy). Również w 1950 r. *C. Shannon* opublikował artykuł o szachach komputerowych³. Wprowadził w nim podział na dwa typy szachowych metod obliczeniowych. Pierwszą grupę stanowią metody brutalne – analizujące wszelkie możliwe posunięcia, a o ich skuteczności decyduje potencjał mocy obliczeniowych wykorzystywanych do ich implementacji. Drugą grupą metod są te, które selekcionują badane sytuacje i nie analizują ich wszystkich. Z punktu widzenia metod sztucznej inteligencji bardziej interesujące są oczywiście programy wykorzystujące strategie selekcyjne. Tego typu program napisał *A. Turing*. W 1952 r. *Alick Glennie* zaimplementował program *A. Turinga* i był pierwszą osobą, która wygrała w szachy z komputerem. Warto podkreślić, że twórcy współczesnej informatyki *A. Turing* i *C. Shannon* interesowali się programami szachowymi, ich tworzeniem i teorią. Polski wybitny matematyk *Stanisław Ulam* na komputerze Univac MANIAC I (11 kHz, pamięć 600 słów) uruchomił program szachowy dla szachownicy 6x6 (dla rozgrywki bez gońców 12 minut potrzebne było dla wykonania ruchu z analizą 4 posunięć w przód). W 1956 r. *John McCarthy* po raz pierwszy użył terminu sztuczna inteligencja – AI (ang. *Artificial Intelligence*). Od tego czasu postęp w dziedzinie programów szachowych związany jest z osiągnięciami sztucznej inteligencji. Często algorytmy opracowywane z myślą o grach znajdują zastosowanie w budowie strategii przeszukiwania dla WWW, w biologii molekularnej, czy w ekonomii. Przykładem jest algorytm cięć α - β opracowany w 1958 r. (*Allen Newell, Cliff Shaw, Herbert Simon*⁴), ograniczający analizę posunięć tylko do sensownych, umożliwiając początkowe analizowanie posunięć najbardziej obiecujących. W 1958 r. program NSS (IBM 704) odniósł pierwsze zwycięstwo w grze przeciwko człowiekowi (osoba grająca poznała zasady gry w szachy godzinę przed rozgrywką). Lata sześćdziesiąte i siedemdziesiąte ubiegłego wieku to czas pojedynków komputerów (głównie amerykańskich z radzieckimi), mistrzostw świata komputerów szachowych i coraz bardziej ciekawych pojedynków programów i komputerów szachowych z wybitnymi graczami. Zwieńczeniem tych dokonań był pojedynek *Gari Kasparowa* z komputerem Deep Blue z 200 procesorami szachowy-

² Osiemnastowieczna maszyna szachowa *von Kempelena* (mechaniczny Turek grający w szachy) była mechanizmem wykonującym ruchy zlecone przez gracza zamkniętego w jej wnętrzu.

³ *C. Shannon*, Programming a computer for playing chess, *Philosophical Magazine* 1950, Series 7, vol. 41, Nr 314.

⁴ *A. Newell, C. Shaw, H. Simon*, Chess Playing Programs and the Problem of Complexity, *IBM Journal of Research and Development* 1958, vol. 4, Nr 2, s. 320–335.

mi. Deep Blue wygrał partię z mistrzem świata (cały mecz skończył się zwycięstwem *Kasparowa* 4–2; Deep Blue analizował ponad 200 milionów pozycji na sekundę, natomiast *Kasparow* analizował 10 pozycji w ciągu 3 minut).

W cieniu spektakularnego boju (szczególnie że dotyczył on również rywalizacji supermocarstw) o pokonanie mistrzów szachowych przez automat, konstruowano programy dla prostszej gry – warcabów⁵. Pierwszy uczący się program gry w warcaby stworzył w 1952 r. *Arthur Lee Samuel*⁶. Był to pierwszy program samouczący się, który stosował różne funkcje i heurystyki dla ograniczenia analizowanych posunięć (właśnie w optymalizacji tych funkcji wartościujących stany gry wykorzystywany był proces samouczenia). Wiele technik i pomysłów *A.L. Samuela* jest wykorzystywanych we współczesnej sztucznej inteligencji⁷. W 1992 r. stworzony przez *Jonathana Schaeffera* program *Chinook* wygrywa US Open w warcaby. Program ten stosował algorytm cięć α - β , ale był wspomagany przez, olbrzymie jak na owe czasy, systemy komputerowe. Na jego potrzeby stworzone zostały bazy danych optymalnych debiutów i końcówek rozgrywki, redukujące czas potrzebny na wartościowanie pozycji. *J. Schaeffer* jest współautorem ciekawego eksperymentu obliczeniowego, który, w pewnym sensie, doprowadził do utraty sensu rozgrywania mistrzostw w warcaby⁸. Przez 18 lat, niemal nieprzerwanie, dziesiątki komputerów pracowało nad rozwiązywaniem warcabów (zaangażowane były wszystkie komputery z Uniwersytetu Alberty, w szczytowym momencie eksperymentu wykorzystywane było 200 komputerów stacjonarnych pracujących nad tym problemem w pełnym wymiarze godzin, a w pozostałych latach zmniejszona liczba do około 50). Baza danych programu *Chinook* zyskała pełną informację o wszystkich 5×10^{20} możliwościach posunięć w warcabach. Twórcy programu udowodnili, że w przypadku, gdy żaden z graczy nie popełni błędu, partia warcabów musi skończyć się remisem. Autorzy tego rozwiązania ogłosili, że w tym wypadku heurystyka została zastąpiona perfekcją. Oczywiście można w tym wypadku argumentować, że sukces programu *Chinook* to sukces „brutalnej siły” – mocy obliczeniowej i pamięciowej komputerów, a nie sztucznej inteligencji (mimo że autorzy programu i eksperymentu stosowali strategię przeszukiwania i wartościowania przypisane do metod sztucznej inteligencji).

⁵ Analizowano warcaby w wersji angielskiej popularne w USA i Wielkiej Brytanii (na szachownicy 64-polowej). W Polsce stosujemy nieco inne zasady – np. bicie do tyłu nie jest zabronione, a damki mogą przesuwac się o więcej niż jedno pole. Istnieje też odmiana sportowa – rozgrywana na planszy o stu polach.

⁶ *A.L. Samuel*, Some Studies in Machine Learning Using the Game of Checkers, *IBM Journal* 1959, Nr 3(3), s. 210–229.

⁷ *L. Bolc, J. Cytowski*, Search Methods for Artificial Intelligence, London 1992.

⁸ *J. Schaeffer, N. Burch, Y. Björnsson, A. Kishimoto, M. Müller, R. Lake, P. Lu, S. Sutphen*, Checkers Is Solved, *Science* 2007, vol. 317 (5844), s. 1518–1522.

3. Turnieje wiedzy – systemy ekspertowe i rozumienie języka naturalnego

Ważnym obszarem zastosowań sztucznej inteligencji są systemy ekspertowe. System ekspertowy to system oparty na wiedzy, który wykorzystuje wiedzę z danej dziedziny i wykorzystuje procedury wnioskowania do rozwiązywania problemów, które wymagają kompetencji lub wiedzy fachowej. Siła systemów ekspertowych wynika przede wszystkim ze zgromadzenia dużej ilości aktualnych informacji i danych z danej dziedziny, przechowywanych w bazie wiedzy systemu ekspertowego. Zazwyczaj systemy ekspertowe są asystentami decydentów, nie zastępcami. Baza wiedzy systemu eksperckiego zawiera również wiedzę heurystyczną, wyznaczoną przy analizie metod i strategii ekspertów pracujących w danej dziedzinie.

Systemy ekspertowe to programy wyspecjalizowane w: udzielaniu porad, analizowaniu, kategoryzacji, konsultacji, projektowaniu, prognozowaniu, wyjaśnianiu, przewidywaniu, zarządzaniu, kontroli, planowaniu i testowaniu. Systemy te często radzą sobie z sytuacjami, w których nie ma pewności dedukcji i logicznego rozumowania. Systemy eksperckie dają możliwość stworzenia głębokiego zrozumienia wiedzy. Często proces ich budowy pozwala pomóc ekspertom świadomie i dokładnie przeanalizować ich praktyczne postępowanie. By systemy ekspertowe były użyteczne dla specjalistów i niespecjalistów konieczna jest jak najmniej sformalizowana komunikacja użytkownika z systemem, najlepiej w języku naturalnym.

W 2011 r. superkomputer IBM Watson wziął udział w amerykańskim konkursie Jeopardy (nasz odpowiednik to *Va Banque*) i wygrał, pokonując swoich dwóch ludzkich przeciwników. Watson (nazwany także *DeepQA*) to tak naprawdę program komputerowy oraz superkomputer stworzony na podstawie innego superkomputera IBM – *Blue Gene/P*. W tym wypadku baza wiedzy była ogromna, w skład całego systemu wchodziła ogromna baza danych, do której autorzy projektu pod przewodnictwem *Davida Ferruccio*⁹ wprowadzili różne encyklopedie, powieści, słowniki, artykuły oraz związki między słowami. Prócz ogromnej bazy wiedzy, Watson to około 100 różnych algorytmów przeszukujących i analizujących bazę danych. Dzięki ogromnej mocy obliczeniowej superkomputera IBM algorytmy nie były wykonywane sekwencyjnie, lecz równolegle. Pozwalało to znacząco skrócić czas reakcji komputera i sprawić, że miał szansę z ludźmi w konkursie, w którym limit czasu ma decydujące znaczenie. Jeśli kilka algorytmów przedstawiało te same wyniki, były ustawiane w specjalnym rankingu. Jeśli żadna z odpowiedzi nie przekraczała ustalonego przez projektantów pułapu procentowego, Watson wstrzymywał się od odpowiedzi, by nie stracić punktów. Warto zaznaczyć, że

⁹ D. Ferrucci, A. Levas, S. Bagchi, D. Gondek, E. T. Muelle, Watson: Beyond Jeopardy!, *Artificial Intelligence* 2013, Nr 199–200, s. 93–105.

system w żaden sposób nie był podłączony do Internetu, korzystał jedynie ze swoich baz danych. Wszystkie pytania były wczytywane Watsonowi w formie tekstowej. Watson nie korzystał z systemu rozpoznawania głosu. Na potrzeby pojedynku w *Va Banque* nie zadawano pytań zawierających materiały audio i wideo. System udzielał swoich odpowiedzi płynnym, syntetycznym głosem.

Watson był ograniczony poziomem procentowym odpowiedzi. Innymi słowy, musiał być pewny jakiejś odpowiedzi w stopniu określonym przez IBM, by udzielić odpowiedzi. Ciekawa rzecz wydarzyła się przy podpowiedzi: „Antyczna rzeźba Lwa z Nimrud zginęła z muzeum narodowego w tym mieście w 2003 r.”. Poprawną odpowiedzią, a w zasadzie pytaniem było: „Czym jest Bagdad?”. Watson był pewny tej odpowiedzi w zaledwie 32%, co było poniżej progu odpowiedzi ustalonego przez inżynierów IBM. Watson przeanalizował jednak możliwą stratę wynikającą z niepodania odpowiedzi i wbrew zasadom wytypował odpowiedź i ... wygrał.

W 2011 r. IBM rozpoczęło prace (z wykorzystaniem systemu Watson, po sukcesie w *Va Banque*) nad praktycznym i komercyjnym zastosowaniem sztucznej inteligencji. W 2014 r. IBM otworzył nową siedzibę dla swojego działu sztucznej inteligencji, znanego jako IBM Watson. Największe nadzieje związane były z budową systemów wykorzystywanych w opiece zdrowotnej.

W jednym z pierwszych pokazów Watson przedstawił, dla szczególnego zbioru objawów pacjenta, listę możliwych diagnoz, każda z adnotacjami poziomu zaufania Watsona i linkami do pomocniczej literatury medycznej. Baza wiedzy Watson'a zawierała informacje o wielu rzadkich chorobach, nie była podatna na rutynę i przyzwyczajenia. Wydawało się, że sztuczna inteligencja mogłaby ograniczyć błędy diagnostyczne, zoptymalizować leczenie, a nawet złagodzić niedobory lekarzy – nie poprzez zastępowanie lekarzy, ale pomaganie im w szybszym i lepszym wykonywaniu pracy.

IBM Watson stworzył narzędzia doradcze dla onkologów i genetyków. Wykorzystywały one przetwarzanie języka naturalnego do analizy elektronicznej dokumentacji medycznej pacjentów, a następnie przeszukiwało bazy danych w celu uzyskania zaleceń dotyczących leczenia. Watson for Oncology zdobywał wiedzę, przyswajając obszerną literaturę medyczną na temat raka oraz analizując dokumentację medyczną prawdziwych pacjentów z rakiem. Największe nadzieje wiązano z wykorzystaniem potężnej mocy obliczeniowej do analizy tych wielkich zbiorów danych (również danych demograficznych, charakterystyk poszczególnych przypadków, przebiegu leczenia i ich rezultatu) i odkrycia wzorców i wiedzy niewidocznej dla lekarzy. Analizowane były dane najświeższe z wieloma artykułami, które są publikowane każdego dnia. Watson dość szybko nauczył się pozyskiwać wiedzę z artykułów dotyczących badań klinicznych, przyswajając ich wyniki. Okazało się jednak, że niemożliwe było nauczanie Watsona czytania artykułów tak, jak zrobiłby to lekarz. Potężna technologia nie może się równać z chaotyczną rzeczywistością dzi-

siejszego systemu opieki zdrowotnej. Próbując zastosować Watson do leczenia raka, jednego z największych wyzwań medycyny, IBM napotkał fundamentalną rozbieżność między sposobem uczenia się maszyn a sposobem pracy lekarzy. Watson miał fenomenalne umiejętności przetwarzania języka naturalnego. Jednak w dokumentacjach medycznych może brakować danych, mogą one być zapisane w niejednoznaczny sposób lub nie w porządku chronologicznym. W standardowych diagnozach dokładność systemu Watson for Oncology wynosiła 90–96%. Jednak w przypadku informacji zależnych od czasu (takich jak ramy czasowe terapii) skuteczność spadała do około 65%. Pojawiły się również publikacje dotyczące porównawczych badań systemu w innych niż amerykański systemach opieki zdrowotnej. Niestety skuteczność systemu Watson oscylowała pomiędzy 49% a 75%. Nieco większe nadzieje związane są z systemem Watson for Genomics. W 2017 r. system ten w przypadku 32% pacjentów z rakiem objętych analizą, zauważył potencjalnie ważne mutacje, które nie zostały zdiagnozowane przez lekarzy. Pacjenci ci rozpoczęli badania kliniczne związane z nowym lekiem.

Pewna porażka, mocno reklamowanych, projektów sztucznej inteligencji w opiece medycznej wiąże się również z obecnymi standardami w diagnostyce i badaniach medycznych. Wniosek systemu sztucznej inteligencji, opierającego swoje analizy na wzorcach wykrytych w dokumentacji medycznej (np., że określony typ pacjenta radzi sobie lepiej z pewnym lekiem) nie zostałby uznany za oparty na dowodach i dopuszczony do wykorzystania. Bez ściśle kontrolowanego badania klinicznego takie odkrycie byłoby uważane tylko za korelację, a nie zalecenie. Standardy opieki zdrowotnej muszą się zmienić, aby sztuczna inteligencja mogła w pełni wykorzystać swój potencjał i przekształcić medycynę. Systemy sztucznej inteligencji mogą uwzględniać o wiele więcej czynników, niż kiedykolwiek będzie reprezentowane w badaniu klinicznym, i mogą podzielić pacjentów na znacznie więcej kategorii (zapewnić terapię bardziej spersonalizowaną). Powinny nastąpić też zmiany infrastrukturalne, instytucje opieki zdrowotnej muszą zgodzić się na udostępnianie swoich danych zastrzeżonych i kontrolowanych przez prywatność, aby systemy sztucznej inteligencji mogły uczyć się na podstawie dokumentacji milionów pacjentów obserwowanych przez wiele lat.

System IBM Watson to obecnie jednak ogólnie dostępne narzędzie¹⁰ przetwarzające nieuporządkowane obrazy, wiadomości e-mail, dane z portali społecznościowych oraz inne informacje, formułujące na ich podstawie kluczowe spostrzeżenia i schematy oraz identyfikujące istniejące pomiędzy nimi związki. Pozwala zdobywać ciekawe informacje na podstawie trendów społecznych w czasie rzeczywistym dzięki rozwiązaniom wykorzystującym przetwarzanie języka naturalnego. Inna usługa systemu umożliwia szybsze wprowadzanie pro-

¹⁰ Zob. <https://www.ibm.com/pl-pl/watson> (dostęp: 14.7.2021 r.).

duktów na rynek oraz budowanie lepszych interakcji z klientami dzięki zrozumieniu ich potrzeb i odpowiedniemu reagowaniu na nie.

Szczególnie interesujące w kontekście osiągnięć w przetwarzaniu języka naturalnego są możliwości analizy tekstu w celu wyodrębnienia z treści metadanych takich jak pojęcia, obiekty i uczucia; przekształcanie dźwięku i głosu w tekst pisany; przekształcanie tekstu pisanego w naturalnie brzmiącą mowę w różnych językach i z możliwością wyboru różnych głosów; analiza tekstu w celu wyodrębnienia z treści metadanych takich jak pojęcia, obiekty i uczucia; analiza emocji i tonów w tekście pisanym; przewidywanie cech osobowości, potrzeb oraz wartości na podstawie tekstu pisanego.

4. Programy grające samouczące się – sieci neuronowe

Prace nad komputerowym modelem mózgu rozpoczęły się w latach czterdziestych XX w. W 1957 r., *Frank Rosenblatt* oraz *Charles Wightman* zaprojektowali i zbudowali sztuczną sieć neuronową zwaną perceptronem. Perceptron był częściowo elektromechanicznym, częściowo elektronicznym urządzeniem rozpoznającym znaki alfanumeryczne. Wykorzystywany był przy tym proces uczenia się jako metoda programowania systemu. Perceptron mógł klasyfikować tylko dane liniowo separowalne, jego możliwości zastosowań praktycznych były mocno ograniczone. Dopiero w połowie lat osiemdziesiątych XX w. rozpoczęły się eksperymenty z zastosowaniem nieliniowych sieci neuronowych, z nowymi algorytmami uczenia sieci, również z algorytmem wstecznej propagacji błędów¹¹. W latach dziewięćdziesiątych początkowy entuzjazm związany z nieliniowymi sieciami neuronowymi zaczął wygasać. Możliwości techniczne sztucznych sieci neuronowych były limitowane przez istniejącą w tych latach technologię komputerową. Dla implementacji sieci zdolnej do stosunkowo nieskomplikowanych obliczeń potrzebna była sieć wielowarstwowa, której uczenie przy dużej liczbie neuronów było obliczeniowo niepraktyczne (tygodnie, miesiące, lub lata obliczeń). Ostatnie lata i postęp technologiczny spowodowały prawdziwą eksplozję zastosowań sieci neuronowych w bardzo wielu dziedzinach praktycznych, a głównie w analizie dużych zbiorów danych.

Strategie samouczenia się odniosły sukces w grach, w których poziom mistrzowski był do tej pory dla automatów trudny – warcaby, szachy lub nieosiągalny – go.

System *Blondie24* (nazywany również *Anakonda*) nauczył się grać w warcaby na poziomie mistrzowskim, wykorzystując strategie genetyczne i sieci neuronowe (strategia genetyczna sterowała zbiorem sieci neuronowych – wypracowanie strategii gry nastąpiło w drodze szeregu partii, w których Blon-

¹¹ *D.E. Rumelhart, G.E. Hinton, R.J. Williams*, Learning representations by back-propagating errors, *Nature* 1986, Nr 323, s. 533–536.

die24 grał sam ze sobą). Jego powstanie jest opisane w książce *Davida Fogela*¹². W przeciwieństwie do wcześniejszych rozwiązań programistycznych i sprzętowych Blondie24 nie miał dostępu do strategii stosowanych przez ludzkich wybitnych graczy ani do baz danych ruchów dla debiutów i końcówek (nie korzystał z wiedzy ekspertów). Niektórzy badacze uważają, że Blondie24 to pierwsza maszyna myśląca. Ten system neuronowo-genetyczny był w stanie pokonać większość (>99%) ludzkich graczy zarejestrowanych w tym czasie na stronie skupiającej graczy z całego świata. Blondie24 nie grał na poziomie Chinook. Celem tego eksperymentu było jednak wykazanie, że możliwe jest skonstruowanie programu, w którym zapisana wiedza początkowa sprowadza się do zasad gry. Późniejsze modyfikacje pozwoliły Blondie24 pokonać program Chinook.

Kilkanaście lat po pokonaniu przez komputer IBM Deep Blue najlepszego gracza w historii szachów *Garry Kasparowa*, najsilniejsze programy grające w go, wykorzystujące techniki sztucznej inteligencji, osiągnęły jedynie amatorski poziom 5 dan i nadal nie mogły pokonać profesjonalnego gracza w go bez handicapów. Od 2014 r. firma DeepMind Technologies (przejęta w późniejszych latach przez Google) rozpoczęła projekt AlphaGo. W jego wyniku powstawały coraz skuteczniejsze programy komputerowe, a celem projektu było zbudowanie programu zdolnego pokonać profesjonalnych graczy w go.

Algorytmy AlphaGo¹³ wykorzystywały wiele strategii sztucznej inteligencji, zarówno te tradycyjne, jak i najnowsze. Algorytmy AlphaGo głębokie uczenie łączyły z przeszukiwaniem techniką MonteCarlo. Sieć neuronowa była uczona w celu identyfikowania najlepszych ruchów. Poprawiała siłę przeszukiwania drzewa. W procesie uczenia wykorzystywana była również wiedza, sugestie graczy oraz wiedza historyczna z rozgrywek. Po osiągnięciu stopnia biegłości na poziomie profesjonalnym, AlphaGo rozpoczął serie pojedynków z najlepszymi graczami. Celem tych gier było dalsze doskonalenie systemu (dla uniknięcia marnowania czasu graczy, system poddawał partię, gdy samoocena prawdopodobieństwa wygranej spadała poniżej ustalonego progu). System miał olbrzymie wsparcie sprzętowe ponad 1000 procesorów i blisko 200 kart graficznych. W 2017 r. AlphaGo pokonał najlepszych graczy na świecie, nagrodzony został profesjonalnym stopniem 9-dan (taki stopień AlphaGo przyznało kilka stowarzyszeń).

W 2017 r. zespół DeepMind AlphaGo otrzymał medal IJCAI Marvin Minsky za wybitne osiągnięcia w sztucznej inteligencji. Komitet Nagród IJCAI wyraził przy tym szczególne uznanie dla systemu AlphaGo za doskonałe połączenie klasycznych technik sztucznej inteligencji z najnowocześniejszymi.

¹² D.B. Fogel, *Blondie24: Playing at the Edge of AI*, Burlington 2002.

¹³ D. Silver, A. Huang, *Mastering the game of Go with deep neural networks and tree search*, Nature 2016, Nr 529(7587), s. 484–489.

szymi technikami uczenia maszynowego. AlphaGo zostało nazwane wspaniałym osiągnięciem współczesnej sztucznej inteligencji.

Druga dekada XXI w. to wykorzystanie wystarczająco szybkich komputerów i wystarczająco dużych zbiorów danych do skutecznego trenowania dużych sieci neuronowych. Duże sieci neuronowe, uczone bez nadzoru, to urzeczywistnienie idei głębokiego uczenia. Są one powszechnie wykorzystywanym narzędziem we współczesnej informatyce.

5. Autonomiczne programy grające, reagujące na zmiany w czasie rzeczywistym – robotyka i systemy agentowe

Agenci, a dokładniej inteligentni agenci, to nowy paradygmat tworzenia systemów sztucznej inteligencji, ukierunkowany na symulację złożonych interakcji. Systemy agentowe to jeden z głównych obszarów zainteresowań informatyki, sztucznej inteligencji i teorii systemów złożonych.

Paradygmat inteligentnych, racjonalnych agentów jest wykorzystywany w wielu dziedzinach. Mogą to być stosunkowo niewielkie systemy, np. filtry poczty e-mail. Taki schemat konstrukcyjny mogą mieć również duże złożone systemy, takie jak kontrola ruchu lotniczego, analizy lotów stad ptaków lub analizy ludzkich zachowań społecznych.

Zanim zajmiemy się kwestią rozwoju systemów agentowych (ABS), należy spróbować zdefiniować, co oznaczają terminy „agent” i ABS. Niestety, brakuje powszechnie akceptowanych definicji kluczowych pojęć w obliczeniach agentowych. W rzeczywistości nie ma prawdziwej zgody co do definicji terminu agent. Definicje systemów agentowych nie są jednoznaczne. Opisując własności takich systemów, trzeba uwypuklić następujące cechy agenta: powinien być możliwy do zidentyfikowania, być dyskretną jednostką z zestawem cech i reguł (matematycznych lub logicznych), które rządzą zachowaniem i podejmowaniem decyzji; powinien być ulokowany w środowisku, z którym oddziałuje i w nim oddziaływać na innych agentów; powinien być zorientowany na cel; powinien być samowystarczalny; powinien być elastyczny i mieć zdolność uczenia się, doskonalenia swojego zachowania poprzez doświadczenia uzyskane w czasie swojego funkcjonowania.

Badania systemów agentowych rozpoczęły się w późnych latach osiemdziesiątych i dziewięćdziesiątych XX w. Od samego początku związane były z teorią gier. Szybko jednak znaleziono inne zastosowania takie jak zautomatyzowane negocjacje (w ramach których badano rozwój protokołów, aukcji i strategii negocjacyjnych umożliwiających agentom pracującym w sieci szybkie negocjowanie bez interwencji człowieka). Pojawiły się zastosowania w handlu elektronicznym i reklamie (np. przy użyciu systemów agentowych prowadzono negocjacje firm przy wyświetlaniu reklam użytkownikom).

Aktualnie dynamicznie rozwijającym się obszarem zastosowań systemów agentowych są gry wideo. Tak jak to już zauważyliśmy, gry wyznaczały środowisko kooperacyjne i konkurencyjne (czasem oba jednocześnie). Obecnie systemy agentowe stosowane są również w samych grach wideo (gry nie są już polem eksperymentów, lecz dzięki agentom są bardziej doskonałe). Sztuczna inteligencja lokowana jest w grach w różny sposób, ale często wykorzystywany jest tu paradygmat agentowy (naturalny w tym przypadku). Agenci mogą wchodzić w interakcje i współpracować w unikalny i interesujący sposób z graczami – ludźmi. Systemy agentowe są także wykorzystywane do doskonalenia dobierania graczy i rankingu w grach wieloosobowych, szczególnie w grach *online*.

Z początkiem XXI w. pojawiło się nowe wyzwanie budowy algorytmów zdolnych rozwiązywać różne zadania z różnych dziedzin. Takiej koncepcji konstruowania systemów i programów przypisano nazwę „dużej sztucznej inteligencji”. Te nowe produkty sztucznej inteligencji mają odzwierciedlać kompetencje ogólne, uczyć się w różnych dziedzinach, planować w wielu obszarach, a w szczególności grać w różnorakie gry. Klasyczne gry wideo okazały się idealnym miejscem do testowania nowych i ulepszonych podejść algorytmicznych, a szczególnie opracowywania zasad nawigacji w złożonych środowiskach (co jest szczególnie ważne w robotyce).

Projektowanie systemów agentowych o różnorodnych kompetencjach rodzi pytanie, jak najlepiej je ocenić, co z kolei przekłada się na zasady ich uczenia. Oceny empiryczne z analizowanych problemów porównawczych są w tym wypadku wadliwe. Powodują nadmierną adaptację do danej domeny zastosowań, uniemożliwiają przeniesienie algorytmu do nowych zastosowań. Dlatego warto testować te nowe algorytmy w domenach, które są wystarczająco zróżnicowane, a zarazem ogólne. Każde z zadań w nich zawartych jest interesujące, a równocześnie reprezentatywne dla zastosowań praktycznych. Każde było przy tym maksymalnie niezależne od pozostałych.

Arcade Learning Environment – ALE¹⁴ to środowisko eksperymentalne dla systemów o kompetencjach ogólnych. ALE to narzędzie do projektowania programów dla ponad 500 gier Atari 2600 (konsola do gier produkowana od 1977 r. przez amerykańską firmę Atari). Jest to środowisko do testowania znanych i nowych algorytmów uczenia się, metod planowania i strategii agentowych. W ostatnich latach pojawia się wiele nowych doniesień o sukcesach na platformie ALE. Na przykład Agent57¹⁵ to system agentowy, wykorzystujący techniki głębokiego uczenia, wygrywający z najlepszymi graczami w 57 gier

¹⁴ M.G. Bellemare, Y. Naddaf, J. Veness, M. Bowling, The Arcade Learning Environment: An Evaluation Platform For General Agents, Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence 2015 (IJCAI 2015), s. 4148–4152.

¹⁵ A.P. Badia, B. Piot, S. Kapturowski, P. Sprechmann, A. Vitvitskiy, D. Guo, Ch. Blundell, Agent57: Outperforming the Atari Human Benchmark, arXiv:2003.13350v1 [cs.LG] 30 Mar 2020 (dostęp: 16.7.2021 r.).

Atari 2600. System ten był w stanie osiągnąć doskonałe wyniki uczenia w tak zróżnicowanym zestawie gier. GoExplore¹⁶ osiągnął sukces w 11 grach (analizowano około 30 miliardów ramek dla tych gier) uważanych za najtrudniejsze. Zgodnie z zapewnieniem projektantów GoExplore będzie bardzo przydatny w usprawnianiu platform robotów, pozwalając im rozwiązywać problemy znacznie szybciej. Ważnym zastosowaniem będzie zastosowanie GoExplore do wyszukiwania słabych punktów systemów automatycznych (m.in. w zapobieganiu kolizji samochodów autonomicznych z pieszymi dla zwiększenia bezpieczeństwa). Mimo tak wspaniałych wyników inne zespoły badawcze pracują nad jeszcze większą uniwersalnością systemów.

Robotyka i sterowanie pojazdami autonomicznymi to najbardziej spektakularne zastosowania współczesnych systemów agentowych. Roboty magazynowe Amazon mają znaczący wpływ na wydajność dostaw, pomagają w ogromnej sprzedaży. Autonomiczne samochody wykorzystują systemy agentowe poprzez symulację działań innych agentów (mogą to być również ludzie). Systemy agentowe są również wykorzystywane do negocjacji ruchu między pojazdami autonomicznymi. Komunikacja na drodze jest naturalnym systemem wieloagentowym, znajdują się tam inni kierowcy w swoich pojazdach, piesi, rowerzyści, również inne autonomiczne samochody. Rozwój pojazdów autonomicznych to prawdziwy konglomerat różnych technologii sztucznej inteligencji. Metody uczenia i wizji komputerowej są koordynowane przez systemy agentowe.

6. Podsumowanie

Kilka przełomowych zdarzeń w rozwoju sztucznej inteligencji, co jak mam nadzieję udało się w niniejszym opracowaniu wykazać, związanych jest z grami. Przedmiotem analizy były głównie gry i problemy deterministyczne z pełną informacją. Równie obszerne opracowanie mogłoby być poświęcone grom z elementami losowymi (tasowanie kart, rzut kostką) lub grom z niepełną informacją (np., gdy gracz nie widzi kart swoich przeciwników). Badania tego rodzaju gier rozpoczęły się później (najciekawsze wyniki dotyczą pokera). Wykorzystywano w projektowaniu programów grających strategię podobne do opisanych w niniejszym opracowaniu, w szczególności głębokie uczenie. W rozgrywce użyteczna jest jednak również psychologia, intuicja, wykorzystanie słabszych umiejętności przeciwnika. Analiza takich gier jest ważna, ponieważ praktyczne strategie z dziedziny ekonomii, zarządzania, planowania składają się często z blefowania, małych taktyk oszustwa, przewidywania działań przeciwnika.

¹⁶ A. Ecoffet, J. Huizinga, J. Lehman, K.O. Stanley, J. Clune, First return, then explore, *Nature* 2021, Nr 590(7847), s. 580–586.

Paradygmaty strategii gier wykorzystywane są w bardzo wielu praktycznych zastosowaniach, obejmujących oprócz ekonomii, zarządzania, sterowania i planowania, również socjologię, politykę, architekturę czy lingwistykę. Dlatego dokonania sztucznej inteligencji w dziedzinie budowy programów dla gier znajdują wiele zastosowań. Wiele problemów czeka jeszcze na rozwiązanie (choć starożytne gry, szachy i go wydają się pokonane), ale nowe rozwiązania pojawiają się coraz szybciej i są bardzo obiecujące. To co w końcu XX w. wydawało się być marzeniem teraz dostępne jest w codziennie używanych smartfonach. Rozpoznawanie mowy, tłumaczenie tekstów, rozpoznawanie twarzy, rozpoznawanie ptaków po nagraniu ich śpiewu, a przede wszystkim wykorzystywanie olbrzymiej wiedzy zgromadzonej w zasobach internetowych to funkcjonalności i aplikacje stosowane powszechnie. Do ich stworzenia przyczyniły się osiągnięcia sztucznej inteligencji, dla których gry były inspiracją.