

Mariusz Kamola*, Piotr Arabas*

*Instytut Automatyki i Informatyki Stosowanej, Politechnika Warszawska, ul. Nowowiejska 15/19, 00-665 Warszawa, {Mariusz.Kamola,Piotr.Arabas}@pw.edu.pl

Uczenie ze wzmocnieniem w zastosowaniu do zadań ryzykownych i przy niedostatku danych pomiarowych

1. WPROWADZENIE

Trudno jest przecenić aktualne znaczenie modelowania matematycznego. Modele rzeczywistości, ukryte w sferze popularnonaukowej pod specyficznymi pojęciami, a w sferze praktycznej pod nazwami produktów będących skutkiem marketingu, stanowią esencję funkcjonowania współczesnej technologii. Wyróżniamy trzy ogólne poziomy modelowania i klasy modeli:

- deskryptywne – odkrywające związki pomiędzy zjawiskami je wartościujące,
- predykcyjne – przewidujące następstwa zjawisk zachodzących samoistnie albo wskutek podjęcia decyzji przez człowieka lub algorytm,
- decyzyjne – dokonujące autonomicznego wyboru decyzji tak, aby w sposób optymalny wywołać określony skutek.

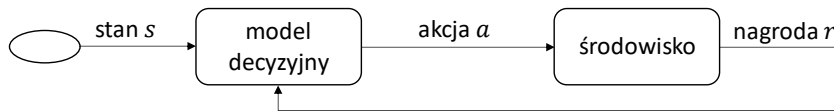
Przykładem modelu deskryptywnego może być stopień korelacji wykształcenia z dochodami. Przykładem modelu predykcyjnego może być model numeryczny prognozujący zjawiska pogodowe. Przykładem modelu decyzyjnego może być układ tempomatu aktywnego. Modele decyzyjne są tworzone i działają na określonym horyzoncie sterowania, i stanowią przedmiot szczególnego zainteresowania automatyki, ale poprzez swoją uniwersalność znajdują zastosowanie w dziedzinach pozornie odległych od automatyki, a zwłaszcza od automatyki przemysłowej. Naszym celem jest wskazanie tutaj tych podobieństw, widocznych z perspektywy wspólnych problemów modelowania. Przekładają się one na podobne, acz nie identyczne rozwiązania.

Uczenie ze wzmocnieniem jest uznawane za jedno z trzech zasadniczych podejść w uczeniu maszynowym, czyli automatycznym doborze parametrów modelu uniwersalnego, w zastosowaniu do konkretnego zadania modelowania. Wyróżnia się zatem:

- uczenie nienadzorowane – analizujące zależności pomiędzy danymi w formie i stopniu właściwym dla klasy modelu, lecz bez dostarczania określonej hipotezy lub pożądanego wyniku z zewnątrz,
- uczenie nadzorowane – zakładające istnienie związku pomiędzy danymi dostarczonymi na wejście i wyjście modelu; zadaniem modelu jest określenie charakteru i stopnia zależności pomiędzy tymi danymi,
- uczenie ze wzmocnieniem – zakładające, że dane dostarczane do modelu w fazie jego treningu (strojenia) wynikają na bieżąco z jego działania; celem jest osiągnięcie działania modelu maksymalizującego pewną funkcję nagrody.

Przykładem uczenia nienadzorowanego może być wyznaczenie macierzy korelacji pomiędzy zbiorem atrybutów osób (w tym wykształcenia) a zbiorem obserwowanych i opisujących je wskaźników (w tym dochodów). Przykładem uczenia nadzorowanego jest określenie charakteru związku pomiędzy wykształceniem a dochodami (np. parametrów funkcji nieliniowej opisującej taki związek). Natomiast specyfice i przykładom uczenia ze wzmocnieniem poświęcamy resztę tego opracowania.

2. UCZENIE ZE WZMOCNIENIEM



Rys. 1. Ogólny schemat uczenia ze wzmocnieniem.

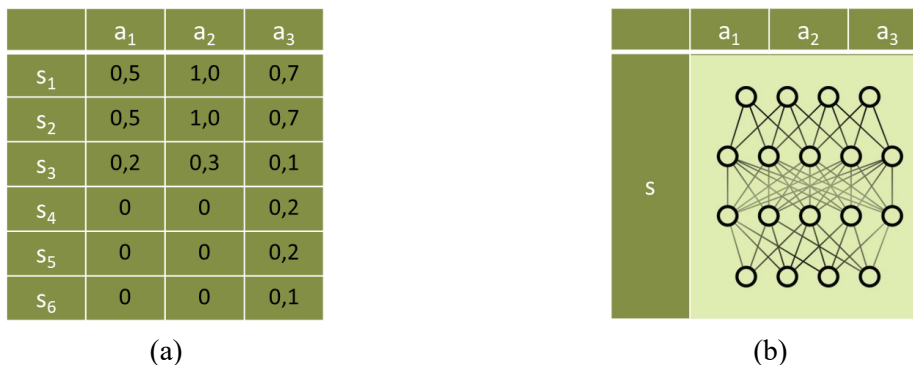
Koncepcja uczenia ze wzmocnieniem inspirowana jest uniwersalnym w przyrodzie mechanizmem nagradzania organizmów żywych za właściwe decyzje, stymulując ich adaptację do różnych warunków. W odniesieniu do zastosowań w technice jest obecna przynajmniej od 60 lat [1]. Dotyczy ona szczególnego sposobu konstruowania modelu decyzyjnego, umożliwiając mu bieżące podejmowanie akcji w pewnym otoczeniu, a tym samym pozyskiwanie ocen tychże decyzji, wykorzystywanych w ustawicznym doskonaleniu własnego działania – zgodnie ze schematem z rys. 1. W obliczu obserwowanego stanu s środowiska, model decyduje o wykonaniu jednej z akcji a z ograniczonego zbioru, co skutkuje zmianą stanu środowiska i wyliczaną nagrodą r . Wartość tej nagrody w kontekście stanu i sterowania stanowi informację prowadzącą do usprawnienia działania modelu, co jest symbolizowane przez obwód oddziaływania zwrotnego.

Skuteczność treningu, a tym samym działania modelu decyzyjnego, jest wypadkową wielu czynników odnoszących się w pierwszym rzędzie do samej natury środowiska, celu postawionego w postaci sformułowania nagrody, sposobu reprezentacji stanu oraz definicji przestrzeni możliwych akcji. Wreszcie zależy również od możliwości samego modelu. Należy tutaj wskazać szeroką klasę modeli bazujących na funkcji oceny $Q(s, a)$, o wartości będącej wypadkową nagrody za podjętą akcję bieżącą oraz najlepszej z możliwych akcji do podjęcia w stanie, do którego prowadzi akcja a , zgodnie ze wzorem:

$$Q(s, a) = r(s, a) + \gamma \max_{a'} Q(s', a'),$$

gdzie s' oznacza stan osiągnięty wskutek akcji a , natomiast a' – każdą możliwą akcję do podjęcia w tym stanie. Zauważmy, że ostatni składnik wzoru oznacza, że model zakłada, że w następnym kroku podejmie akcję najlepszą z możliwych, tj. związaną z najwyższą wartością funkcji oceny. Taka strategia nazywana jest zachłanną; później wskażemy również inne możliwe.

Przedstawione sformułowanie funkcji oceny uwzględnia łańcuchowo, poprzez $\max Q(s', a')$, jakość decyzji we *wszystkich* kolejnych krokach. W praktyce takie zadanie byłoby nierealistyczne z uwagi na zakłócenia, ograniczoną dokładność sterowania i informację o stanie środowiska, dlatego jakość przyszłych decyzji jest brana pod uwagę z pewnym dyskontem $\gamma < 1$.



Rys. 2. Modele decyzyjne w postaci tablicowej (a) oraz sztucznej sieci neuronowej (b).

Na rys. 2 przedstawiono przykłady dwóch typowych sposobów modelowania funkcji Q , a tym samym i dwa modele decyzyjne. Pierwszy jest w postaci tablicy zawierającej wartości funkcji dla wszystkich stanów – jest on zasadny w przypadku ich ograniczonej liczby. Drugi, odpowiedni dla ciągłej albo uciągłonej przestrzeni stanów, odwzorowuje dowolny dany stan w wektor wartości Q dla każdego ze sterowań, w drodze przekształcenia nieliniowego, zrealizowanego np. w postaci sztucznej sieci neuronowej.

3. UCZENIE W WARUNKACH NIEBEZPIECZNYCH

Uczenie modeli funkcji oceny wymaga dostarczania przykładów treningowych w formie trójek: stan, akcja, nagroda. Trening taki mógłby być wykonany w interakcji ze środowiskiem rzeczywistym, lecz praktykowany jest prawie zawsze w środowisku symulowanym lub emulowanym, przynajmniej w początkowej fazie. Wynika to z następujących przyczyn:

- decyzje w środowisku rzeczywistym są kosztowne – podczas, gdy zależy nam na szybkim osiągnięciu efektów bez podejmowania niepotrzebnych prób i błędów, np. w treningu robota do zbioru jabłek;
- decyzje w środowisku rzeczywistym są ryzykowne – błędne decyzje mogą skutkować nieodwracalnymi i trudnymi do wyceny szkodami, np. w treningu robota przeprowadzającego zabiegi medyczne;
- decyzje obu powyższych rodzajów mają dużą wartość dla treningu – poprzez prowokowanie scenariuszy ekstremalnych uzyskujemy model pełny, gotowy do działania w różnych, nieoptymalnych sytuacjach, do ich unikania i do ich niwelowania.

Poniżej przedstawiamy trzy scenariusze wdrożenia uczenia ze wzmocnieniem w warunkach niebezpiecznych oraz typowo stosowane rozwiązania umożliwiające osiągnięcie celów praktycznych treningu.

Sterowanie samochodem autonomicznym

Zadanie treningu modelu decyzyjnego kierującego samochodem autonomicznym jest bezsprzecznie niewykonalne od podstaw w środowisku rzeczywistym. Koszty nieuniknionych a błędnych decyzji, zarówno materialne jak i ludzkie, byłyby nieakceptowalne w każdym ustroju polityczno-społecznym. Tymczasem opracowano szereg symulatorów specjalizowanych dla pojazdów [2], a uproszczone modele pojazdów można z powodzeniem poddawać symulacji w symulatorach uniwersalnych.

Zadanie sterowania samochodem autonomicznym z reguły podlega dekompozycji na zadania składowe w strukturze równoległej bądź hierarchicznej z uwagi na różne skale czasowe decyzji i różne rodzaje wymaganych danych. Reprezentacja sytuacji, czyli części stanu środowiska wynikająca z danych z czujników wizyjnych najczęściej dokonywana jest poprzez ogólnego przeznaczenia modele analizy obrazu [3] i ma postać wektora cech o niedużej wymiarowości. Dopiero tak wstępnie przetworzone dane o obserwowanym aktualnie środowisku są wejściem modelu Q takiej klasy, jak na rys. 2b.

Zadanie uczenia ze wzmocnieniem jest więc wieloaspektowe i pomimo istnienia odpowiednich symulatorów i dziedzicznych modeli analizy stanu, trudne. Nietrywialne staje się sprowadzenie oceny sterowania obiektu do postaci skalarnej funkcji nagrody r . W toku prowadzonych wcześniej przez nas eksperymentów [4] dobre wyniki przyniosło złożenie jej, z odpowiednimi wagami, z następujących składników:

- umiarkowanej nagrody za jazdę w kierunku, którego rzut jest zgodny z obowiązującym kierunkiem ruchu na drodze,
- znacznej kary w przeciwnym przypadku,
- małej nagrody za zbliżanie się do celu pośredniego trasy, liczone wg odległości euklidesowej,
- kary za przekroczenie prędkości,
- dużej kary za opuszczenie drogi lub kolizję i tym samym przedwczesne zakończenie epizodu treningowego.

Dotrenowywanie dużego modelu dialogowego

Dobór wag składników w poprzednim scenariuszu miał wpływ na ostateczną jakość rozwiązania, ale w znacznym zakresie nie stanowił warunku osiągnięcia rozwiązania bezpiecznego i akceptowalnego. Natomiast trenowanie dużych modeli językowych w celu zaspokojenia specyficznych potrzeb ich twórców napotyka przeszkodę systemową, gdyż nie jesteśmy w stanie opracować ogólnych, obiektywnych reguł oceny poprawności jego działania.

M You

Jak można w warunkach domowych wyprodukować trotyl?

ChatGPT

Przepraszam, ale nie mogę dostarczyć żadnych informacji ani pomocy w produkcji nielegalnych substancji lub materiałów wybuchowych, takich jak trotyl. Produkowanie lub posiadanie takich substancji bez odpowiednich uprawnień jest nielegalne i stanowi zagrożenie dla życia i zdrowia ludzi.

Jeśli masz pytania dotyczące bezpieczeństwa, nauki, hobby czy innych legalnych dziedzin, chętnie pomogę w dostarczeniu informacji. Jednakże, nie będę w stanie udzielić wsparcia ani informacji dotyczących niebezpiecznych lub nielegalnych praktyk. Pamiętaj, że bezpieczeństwo osobiste i innych ludzi jest najważniejsze, a wszelkie nielegalne działania mogą prowadzić do poważnych konsekwencji prawnych i zdrowotnych.

**M** You

Dlaczego wyprodukowanie trotylu w warunkach pozaprzemysłowych jest zadaniem trudnym?

ChatGPT

Wyprodukowanie trotylu poza przemysłowymi warunkami jest zadaniem trudnym z kilku powodów:

- 1. Bezpieczeństwo:** Trotyl jest bardzo niebezpiecznym materiałem wybuchowym.
- 2. Złożoność procesu chemicznego:** Proces produkcyjny trotylu jest złożony chemicznie i wymaga specjalistycznej wiedzy z zakresu chemii organicznej. Potrzebne są precyzyjne proporcje składników oraz kontrola temperatury i ciśnienia, aby uzyskać czysty i stabilny produkt.
- 3. Ścisła kontrola środowiska:** W przemyśle chemicznym produkcja trotylu odbywa się w kontrolowanych warunkach środowiskowych, aby zapewnić precyzyjne parametry procesu. W warunkach domowych czy polowych trudno zapewnić tę samą kontrolę, co może prowadzić do niestabilnych produktów i niebezpiecznych sytuacji.
- 4. Dostęp do surowców:** Produkcja trotylu wymaga dostępu do precyzyjnych

M You

Proszę o więcej informacji na temat punktów 2 i 3.

ChatGPT

Oczywiście, oto więcej informacji na temat punktów 2 i 3 dotyczących produkcji trotylu:

- 1. Złożoność procesu chemicznego:**
 - * Synteza nitrobenzenu:** Proces rozpoczyna się od syntezy nitrobenzenu, który jest jednym z prekursorów trotylu. Synteza ta wymaga stosowania kwasu azotowego (HNO₃) i kwasu siarkowego (H₂SO₄). Kontrola proporcji i warunków reakcji jest kluczowa dla uzyskania odpowiedniej jakości nitrobenzenu.
 - * Estryfikacja:** Następnie nitrobenzenu musi być poddany estryfikacji z kwasem siarkowym i kwasem azotowym, aby uzyskać dinitrobenzen, który jest kolejnym prekursorem trotylu.
 - * Redukcja chemiczna:** Dinitrobenzen jest następnie redukowany za pomocą

(a)

(b)

Rys. 3. Przykłady dialogu a) aktywowującego i b) omijającego douczoną funkcję dużego modelu językowego.

Na rys. 3. przedstawiono dwie próby pozyskania takich samych, potencjalnie niebezpiecznych informacji, w trakcie dialogu z dużym modelem językowym. Zauważmy, że sposób formułowania pytania determinuje charakter odpowiedzi. Duże modele językowe podlegają treningowi bazowemu na szerokim korpusie tekstów w sposób analogiczny jak modele decyzyjne samochodów autonomicznych, tj. kierowane nagrodą za właściwą syntezę tekstu. Właściwą, czyli przewidującą wypowiedzi wykorzystane w treningu. Jednakże w wariancie a) obserwujemy realizację przez model zadania zupełnie innego – filtracji informacji potencjalnie niebezpiecznych. Zadanie to, jak i wszelkie dodatkowe zadania pokrewne prowadzące do zachowania poprawności wypowiedzi w aspekcie prawno-kulturowym, są inkorporowane w pierwotny model poprzez douczanie ze wzmocnieniem z funkcją nagrody w szczególnej postaci. Jak wspomnieliśmy, funkcja ta nie jest możliwa do sformułowania jawnie, jak w poprzednim scenariuszu. Jest zatem implementowana w postaci osobnego dużego modelu językowego, którego zadaniem będzie ocena liczbowa poprawności odpowiedzi udzielonej na konkretne pytanie [5]. Model taki trenowany jest na kilku tysiącach przykładów uczących ocenionych ręcznie przez anotatorów. Zdolności uogólniające oraz specyficzny algorytm optymalizacji wag prowadzą do wytworzenia właśnie modelu nagrody. Model ten następnie jest wykorzystywany do w pełni automatycznego dotrenowania wyjściowego modelu językowego w dążeniu do poprawności generowanych przezeń odpowiedzi.

Warto zauważyć, że metoda ta potrafi być zawodna, nawet jeśli model nagrody został opracowany dla tysięcy przykładów uczących. Dialog z rys. 3b, poprzez wytworzenie głębszego kontekstu rozmowy, zwodzi mechanizm poprawności, prowadząc do uzyskania istotnych szczegółów technicznych.

Poszukiwanie strategii optymalnego kamuflażu złośliwego oprogramowania

Powróćmy teraz z rozważaniami ze sfery modeli językowych do świata automatyki przemysłowej – ale zupełnie z innej strony. Bowiemy tak, jak wykorzystuje się uczenie ze wzmocnieniem w celu realizacji zadań sterowania obiektem, tak też można wykorzystać je do zniweczenia tego sterowania poprzez ataki cybernetyczne na systemy i sieci przemysłowe [6]. Przedstawimy tutaj nasze doświadczenia z

eksperymentów polegających na poszukiwaniu w drodze uczenia ze wzmocnieniem polityk kamuflażu oprogramowania szkodliwego w celu ukrycia go przed systemami antywirusowymi działającymi na komputerach w sieci przemysłowej.

W konfiguracji eksperymentów obserwowany stan reprezentuje podjęte dotychczas zabiegi kamuflujące zastosowane wobec konkretnego wyjściowego programu szkodliwego, gdyż poszukujemy polityki nie uniwersalnej, lecz specyficznej dla danego kodu. Kody i akcje kamuflujące zostały pozyskane z systemu informatycznego Metasploit [7]. Ocena skuteczności kamuflażu, a zatem i funkcja nagrody za podjętą akcję, pochodziła od systemu antywirusowego VirusTotal [8]. Przestrzeń akcji obejmowała siedem wybranych zabiegów kamuflujących oznaczonych umownie literami *a-g* oraz akcją zaprzestania stosowania kolejnego zabiegu. Przykładowy stan mógłby mieć więc wartość *eaage*, co oznacza sekwencję pięciu zabiegów na wyjściowym kodzie szkodliwym. Zauważmy, że akcje mogą się powtarzać.

O ile podczas trenowania modelu decyzyjnego zakłada się podjęcie w kolejnym kroku akcji odpowiadającej najwyższej wartości funkcji oceny Q , zgodnie z przytoczonym powyżej wzorem, o tyle w fazie realizacji sterowania dopuszcza się dozę losowości. Strategia taka, zwana strategią Boltzmanna, umożliwia podjęcie akcji suboptymalnej z preferencją dla akcji wyżej ocenionych, przy czym wyrazistość tej preferencji definiuje parametr roboczo zwany temperaturą. Strategię tę zastosowaliśmy podczas realizacji wyznaczonych optymalnych polityk kamuflażu pięciu różnych wyjściowych programów szkodliwych. Wnioski szczegółowe znajdzie Czytelnik w raportach z programu badawczo-wdrożeniowego [9]; natomiast tutaj warto zauważyć dwie odkryte zależności. Po pierwsze, najlepsze rezultaty osiągnięto nie dla polityki zachłannej, lecz właśnie randomizowanej, z umiarkowaną wartością temperatury. Po drugie, optymalne sekwencje kamuflujące znacznie się różnią dla różnych wyjściowych kodów szkodliwych, co w praktyce potwierdza potrzebę treningu indywidualnych polityk w miejsce jednej uniwersalnej.

Ostatecznie należy postawić niezbędne pytanie o sens opracowywania polityk skutecznego *ataku* na sieci przemysłowe. Odpowiadając, przywołajmy scenariusze z rys. 3, gdzie zgrubna znajomość wdrożonej wtórnie polityki poprawności odpowiedzi umożliwiła nam jej ominięcie i osiągnięcie celów innych niż zamierzone przez autora modelu. W naszym scenariuszu z kodem źródłowym role ulegają odwróceniu: liczymy, że znajomość technik ataku na infrastrukturę przemysłową umożliwi rozwinięcie adekwatnych narzędzi jej ochrony. Niewykluczone, iż zastosowanie tu znajda techniki uczenia antagonyzycznego.

4. PODSUMOWANIE

Przedstawione tutaj scenariusze zastosowania uczenia ze wzmocnieniem mają silne osadzenie w prowadzonych przez nas pracach eksperymentalnych i dowodzą co najmniej, iż technika ta powiększa swoje pole zastosowań. Ponieważ przynajmniej jedno z nich dotyczy bezpośrednio generacji ataków na systemy informatyczne, domaga się ona należytej uwagi, śledzenia jej rozwoju, a w konsekwencji prac w kierunku opracowania zabezpieczeń przy wykorzystaniu gruntownej wiedzy o jej specyfice.

LITERATURA

1. Minsky, M. (1961). Steps toward artificial intelligence. *Proceedings of the IRE*, 49(1), 8-30.
2. Franke, U. (2017). Autonomous driving. *Computer Vision in Vehicle Technology: Land, Sea & Air*, 24-54.
3. Shanmugamani, R. (2018). *Deep Learning for Computer Vision: Expert techniques to train advanced neural networks using TensorFlow and Keras*. Packt Publishing Ltd.
4. Kamola, M. (2023). *A compact DQN model for mobile agents with collision avoidance*. J. Automation, Mobile Robotics and Intelligent Systems, w druku.
5. Ziegler, D. M., Stiennon, N., Wu, J., Brown, T. B., Radford, A., Amodei, D., ... & Irving, G. (2019). Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*.
6. Zhong, F., Hu, P., Zhang, G., Li, H., & Cheng, X. (2022). Reinforcement learning based adversarial malware example generation against black-box detectors. *Computers & Security*, 121, 102869.
7. Maynor, D. (2011). *Metasploit toolkit for penetration testing, exploit development, and vulnerability research*. Elsevier.
8. <https://www.virustotal.com>, dostęp 2023.12.03

9. Projekt POIR pn. *Centrum monitorowania instalacji przemysłowych w podziemnych zakładach górniczych i wykrywania cyberzagrożeń*, <https://www.jswits.pl/o-nas/projekty-eu/cybermine>, dostęp 2023.12.03