Uniwersalna inteligencja algorytmiczna: podejście matematyczne od góry do dołu

Sekwencyjna teoria decyzji formalnie rozwiązuje problem racjonalnych agentów w niepewnych światach, jeśli znany jest prawdziwy środowiskowy rozkład prawdopodobieństwa a priori. Teoria uniwersalnej indukcji Solomonoffa formalnie rozwiązuje problem przewidywania sekwencji dla nieznanego rozkładu a priori. Łączymy oba pomysły i otrzymujemy bezparametrową teorię uniwersalnej sztucznej inteligencji. Przedstawiamy mocne argumenty, że wynikowy model AIXI jest najinteligentniejszym nieuprzedzonym agentem, jaki jest możliwy. Opisujemy, w jaki sposób model AIXI może formalnie rozwiązać szereg klas problemów, w tym przewidywanie sekwencji, gry strategiczne, minimalizację funkcji, wzmocnienie i uczenie nadzorowane. Główną wadą modelu AIXI jest to, że jest on nieobliczalny. Aby przezwyciężyć ten problem, konstruujemy zmodyfikowany algorytm AIXItl, który jest nadal skutecznie bardziej inteligentny niż jakikolwiek inny agent ograniczony czasem t i długością l. Czas obliczeń AIXItl jest rzędu t·2l. Dyskusja obejmuje formalne definicje relacji porządku inteligencji, problem horyzontu i relacje teorii AIXI z innymi podejściami do sztucznej inteligencji.

Wprowadzenie

Ta sekcja stanowi wprowadzenie do matematycznej teorii inteligencji. Przedstawiamy model AIXI, bezparametrowego, optymalnego agenta uczenia się przez wzmacnianie, osadzonego w dowolnym nieznanym środowisku. Naukę o sztucznej inteligencji (AI) można zdefiniować jako budowę inteligentnych systemów i ich analizę. Naturalną definicją systemu jest wszystko, co ma strumień wejściowy i wyjściowy. Inteligencja jest bardziej skomplikowana. Może mieć wiele twarzy, takich jak kreatywność, rozwiązywanie problemów, rozpoznawanie wzorców, klasyfikacja, uczenie się, indukcja, dedukcja, budowanie analogii, optymalizacja, przetrwanie w środowisku, przetwarzanie języka, wiedza i wiele innych. Jednak formalna definicja obejmująca każdy aspekt inteligencji wydaje się trudna. Większość, jeśli nie wszystkie znane aspekty inteligencji można sformułować jako ukierunkowane na cel lub, dokładniej, jako maksymalizujące jakąś funkcję użyteczności. Dlatego też wystarczy zbadać ukierunkowaną na cel AI; np. (biologicznym) celem zwierząt i ludzi jest przetrwanie i rozprzestrzenianie się. Celem systemów AI powinno być bycie użytecznym dla ludzi. Problem polega na tym, że poza szczególnymi przypadkami nie znamy z góry ani funkcji użyteczności, ani środowiska, w którym agent będzie działał. Te problemy ma rozwiązać teoria matematyczna, ukuta jako AIXI. Załóżmy, że dostępne są nieograniczone zasoby obliczeniowe. Pierwszą ważną obserwacją jest to, że nie czyni to problemu AI trywialnym. Optymalna gra w szachy lub rozwiązywanie problemów NP-zupełnych staje się trywialne, ale prowadzenie samochodu lub przetrwanie w naturze nie. Dzieje się tak, ponieważ samo w sobie jest wyzwaniem, aby dobrze zdefiniować te ostatnie problemy, nie wspominając o przedstawieniu algorytmu. Innymi słowy, problem AI nie został jeszcze dobrze zdefiniowany. Można postrzegać AIXI jako sugestię takiej matematycznej definicji AI. AIXI to uniwersalna teoria sekwencyjnego podejmowania decyzji, podobna do słynnej uniwersalnej teorii indukcji Solomonoffa. Solomonoff wyprowadził optymalny sposób przewidywania przyszłych danych, biorąc pod uwagę wcześniejsze spostrzeżenia, pod warunkiem, że dane są próbkowane z obliczalnego rozkładu prawdopodobieństwa. AIXI rozszerza to podejście na optymalnego agenta podejmującego decyzje osadzonego w nieznanym środowisku. Głównym pomysłem jest zastąpienie nieznanego rozkładu środowiskowego μ w równaniach Bellmana odpowiednio uogólnionym uniwersalnym rozkładem Solomonoffa ξ. Przestrzeń stanów jest przestrzenią kompletnych historii. AIXI jest uniwersalną teorią bez regulowanych parametrów, nie przyjmującą żadnych założeń dotyczących środowiska, poza tym, że jest ono próbkowane z rozkładu obliczalnego. Z perspektywy złożoności algorytmicznej model AIXI uogólnia optymalną pasywną indukcję uniwersalną na przypadek agentów aktywnych. Z perspektywy teorii decyzji AIXI jest sugestią nowego (niejawnego) algorytmu „uczącego się”, który może

przezwyciężyć wszystkie (oprócz obliczeniowych) problemy poprzednich algorytmów uczenia się przez wzmacnianie. Istnieją silne argumenty, że AIXI jest najinteligentniejszym, możliwym agentem bezstronnym. Opisujemy dla szeregu klas problemów, w tym przewidywanie sekwencji, gry strategiczne, minimalizację funkcji, wzmacnianie i uczenie nadzorowane, w jaki sposób model AIXI może je formalnie rozwiązać. Główną wadą modelu AIXI jest to, że jest on nieobliczalny. Aby przezwyciężyć ten problem, konstruujemy zmodyfikowany algorytm AIXItl, który jest nadal efektywnie bardziej inteligentny niż jakikolwiek inny agent ograniczony czasem t i długością l. Czas obliczeń AIXItl jest rzędu $t \cdot 2^l$. Inne omawiane tematy to formalna definicja relacji porządku inteligencji, problem horyzontu i relacje teorii AIXI do innych podejść AI. Niniejszy rozdział ma być delikatnym wprowadzeniem i omówieniem modelu AIXI. Ta sekcja zawiera również odniesienia do wprowadzających podręczników i oryginalnych publikacji na temat algorytmicznej teorii informacji i teorii decyzji sekwencyjnych.

Sekcja 2 przedstawia teorię decyzji sekwencyjnych w bardzo ogólnej formie (zwanej modelem AIμ), w której działania i percepcje mogą zależeć od dowolnych zdarzeń z przeszłości. Wyjaśniamy związek z równaniami Bellmana i omawiamy parametry drugorzędne, w tym (rozmiar) przestrzeni I/O i czas życia agenta oraz ich uniwersalny wybór, który mamy na myśli. Optymalność AIμ jest oczywista z konstrukcji.

Sekcja 3: Jak i w jakim sensie indukcja jest w ogóle możliwa, była przedmiotem długich kontrowersji filozoficznych. Najważniejsze z nich to zasada Epikura dotycząca wielokrotnych wyjaśnień, brzytwa Ockhama i teoria prawdopodobieństwa. Solomonoff elegancko połączył wszystkie te aspekty w jedną formalną teorię wnioskowania indukcyjnego opartą na uniwersalnym rozkładzie prawdopodobieństwa ξ, który jest ściśle związany ze złożonością Kołmogorowa K(x), długością najkrótszego programu obliczającego x. Można pokazać szybką zbieżność ξ do nieznanego prawdziwego rozkładu środowiskowego μ i ścisłe granice strat dla dowolnych ograniczonych funkcji strat i skończonego alfabetu. Można również pokazać optymalność Pareto ξ w tym sensie, że nie ma innego predyktora, który działa lepiej lub równie dobrze we wszystkich środowiskach, a ściślej lepiej w co najmniej jednym. W świetle tych wyników można uczciwie powiedzieć, że problem przewidywania sekwencji posiada uniwersalnie optymalne rozwiązanie.

Sekcja 4: W przypadku aktywnym algorytmy uczenia się przez wzmacnianie są zwykle używane, jeśli μ jest nieznane. Mogą odnieść sukces, jeśli przestrzeń stanu jest mała lub została skutecznie zmniejszona przez techniki generalizacji. Algorytmy działają tylko w ograniczonych (np. markowskich) domenach, mają problemy z optymalnym wyborem między eksploracją a eksploatacją, mają nieoptymalną szybkość uczenia się, są podatne na rozbieżności lub są w inny sposób ad hoc. Formalne rozwiązanie zaproponowane tutaj polega na uogólnieniu uniwersalnego wcześniejszego ξ Solomonoffa, aby uwzględnić warunki działania i zastąpić μ przez ξ w modelu AIμ, co skutkuje modelem AIξ≡AIXI, który twierdzimy, że jest uniwersalnie optymalny. Badamy, czego możemy oczekiwać od uniwersalnie optymalnego agenta i wyjaśniamy znaczenie słów uniwersalny, optymalny itd. Inne omawiane tematy to formalne definicje relacji porządku inteligencji, problem horyzontu i optymalność Pareto AIXI.

Sekcja 5: Pokazujemy, jak wiele klas problemów AI pasuje do ogólnego modelu AIXI. Obejmują one przewidywanie sekwencji, gry strategiczne, minimalizację funkcji i uczenie nadzorowane. Najpierw formułujemy każdą klasę problemów w jej naturalny sposób (dla znanego μ), a następnie konstruujemy formułę w modelu AIμ i pokazujemy ich równoważność. Następnie rozważamy konsekwencje zastąpienia μ przez ξ. Głównym celem jest zrozumienie, w jakim sensie problemy są rozwiązywane przez AIXI.

Sekcja 6: Główną wadą AIXI jest to, że jest on nieobliczalny, lub dokładniej, jest on obliczalny tylko asymptotycznie, co sprawia, że implementacja jest niemożliwa. Aby przezwyciężyć ten problem,

konstruujemy zmodyfikowany model AIXItl, który jest nadal lepszy od każdego innego algorytmu ograniczonego czasem t i długością l. Czas obliczeniowy AIXItl jest rzędu $t \cdot 2^l$. Rozwiązanie wymaga implementacji logiki pierwszego rzędu, definicji uniwersalnej maszyny Turinga w jej obrębie oraz systemu teorii dowodów.

Sekcja 7: Na koniec omawiamy i komentujemy niektóre w inny sposób niewspomniane tematy o ogólnym zainteresowaniu. Zwracamy uwagę na różne tematy, w tym na współbieżne działania i percepcje, wybór przestrzeni I/O, przetwarzanie zaszyfrowanych informacji i osobliwości agentów ucieleśniających śmiertelników. Kontynuujemy spojrzenie na dalsze badania, w tym na optymalność, zmniejszanie skali, implementację, aproksymację, elegancję, dodatkową wiedzę i szkolenie AIXI(tl). Dołączamy również pewne (osobiste) uwagi na temat fizyki nieobliczalnej, liczby mądrości Ω i świadomości.

Agenci w znanych środowiskach probabilistycznych

Ogólne ramy dla AI można postrzegać jako projektowanie i badanie inteligentnych agentów. Agent to system cybernetyczny z pewnym stanem wewnętrznym, który działa z wyjściem y_k na pewne środowisko w cyklu k, odbiera pewne wejście x_k ze środowiska i aktualizuje swój stan wewnętrzny. Następnie następuje kolejny cykl. Dzielimy wejście x_k na regularną część ok i nagrodę r_k, często nazywaną wzmacniającym sprzężeniem zwrotnym. Od czasu do czasu środowisko zapewnia agentowi nagrodę różną od zera. Zadaniem agenta jest maksymalizacja jego użyteczności, zdefiniowanej jako suma przyszłych nagród. Środowisko probabilistyczne można opisać za pomocą prawdopodobieństwa warunkowego μ dla wejść $x_1...x_n$ do agenta pod warunkiem, że agent wyprowadza $y_1...y_n$. Większość, jeśli nie wszystkie środowiska, są tego typu. Podajemy formalne wyrażenia dla wyjść agenta, które maksymalizują całkowitą sumę μ-oczekiwanej nagrody, zwaną wartością. Ten model nazywa się modelem AIμ. Ponieważ każdy problem AI można sprowadzić do tej formy, problem maksymalizacji użyteczności jest zatem formalnie rozwiązywany, jeśli μ jest znane. Ponadto badamy pewne szczególne aspekty modelu AIμ. Wprowadzamy rozkłady prawdopodobieństwa faktoryzowane, opisujące środowiska z niezależnymi epizodami. Występują one w kilku klasach problemów badanych w sekcji 5 i są szczególnym przypadkiem bardziej ogólnych rozdzielnych rozkładów prawdopodobieństwa. Wyjaśniamy również związek z równaniami Bellmana teorii decyzji sekwencyjnych i omawiamy podobieństwa i różnice. Omawiamy drobne parametry naszego modelu, w tym (rozmiar) przestrzeni wejściowych i wyjściowych X i Y oraz czas życia agenta i jego uniwersalny wybór, który mamy na myśli. W tej sekcji nie ma nic niezwykłego; jest to istota teorii decyzji sekwencyjnych, przedstawiona w nowej formie. Notacja i wzory potrzebne w późniejszych sekcjach są po prostu rozwijane. Pozostają dwa główne problemy: problem nieznanego prawdziwego rozkładu prawdopodobieństwa μ, który jest rozwiązany w sekcji 4 oraz aspekty obliczeniowe.

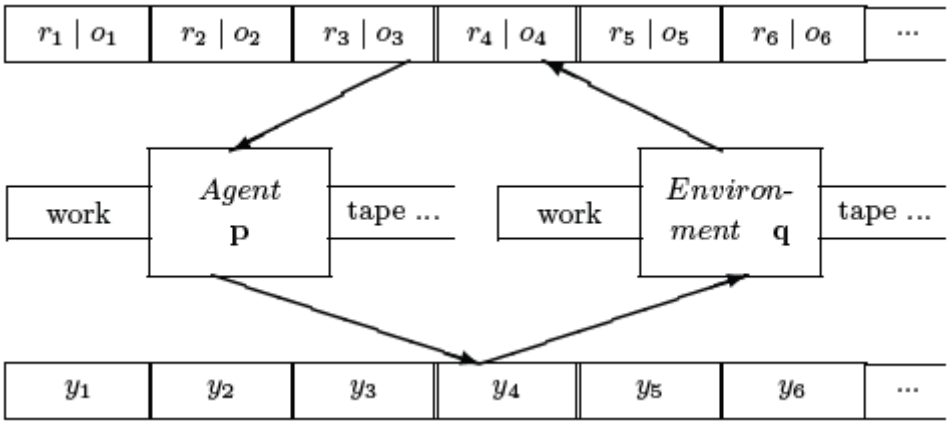
Model agenta cybernetycznego

Dobrym sposobem na rozpoczęcie myślenia o inteligentnych systemach jest rozważenie bardziej ogólnie systemów cybernetycznych, zwykle nazywanych agentami w AI. Pozwala to uniknąć zmagania się ze znaczeniem inteligencji od samego początku. System cybernetyczny to obwód sterujący z wejściem y i wyjściem x oraz stanem wewnętrznym. Na podstawie zewnętrznego wejścia i stanu wewnętrznego agent oblicza deterministycznie lub stochastycznie wyjście. To wyjście (działanie) modyfikuje środowisko i prowadzi do nowego wejścia (percepcji). Trwa to w nieskończoność lub przez skończoną liczbę cykli.

Definicja 1 (model agenta). Agent to system, który wchodzi w interakcje ze środowiskiem w cyklach k = 1,2,3,.... W cyklu k działanie (wyjście) $y_k \in Y$ agenta jest określone przez politykę p, która zależy od historii I/O $y_1x_1...y_{k-1}x_{k-1}$. Środowisko reaguje na tę akcję i prowadzi do nowego postrzegania (danych

wejściowych) $x_k \in X$ określonego przez deterministyczną funkcję q lub rozkład prawdopodobieństwa μ, który zależy od historii $y_1x_1...y_{k-1}x_{k-1}y_k$. Następnie rozpoczyna się następny cykl k+1.

Jak wyjaśniono w poprzedniej sekcji, potrzebujemy pewnego przydziału nagrody do systemu cybernetycznego. Dane wejściowe x są podzielone na dwie części, standardowe dane wejściowe o i pewne dane wejściowe nagrody r. Jeśli dane wejściowe i wyjściowe są reprezentowane przez ciągi znaków, deterministyczny system cybernetyczny może być modelowany przez maszynę Turinga p, gdzie p jest nazywane polityką agenta, która określa (re)akcję na postrzeganie. Jeśli środowisko jest również obliczalne, może być również modelowane przez maszynę Turinga q. Interakcję agenta ze środowiskiem można zilustrować następująco:



Zarówno p, jak i q mają jednokierunkowe taśmy wejściowe i wyjściowe oraz dwukierunkowe taśmy robocze. To, co splątuje agenta z otoczeniem, to fakt, że górna taśma służy jako taśma wejściowa dla p, a także taśma wyjściowa dla q, a dolna taśma służy jako taśma wyjściowa dla p, a także taśma wejściowa dla q. Ponadto głowica odczytująca musi zawsze znajdować się po lewej stronie głowicy piszącej, tj. symbole muszą zostać najpierw zapisane, zanim zostaną odczytane. Zarówno p, jak i q mają własne, wzajemnie niedostępne taśmy robocze zawierające własne „sekrety". Głowice poruszają się w następujący sposób. W k-tym cyklu p zapisuje y_k, q odczytuje y_k, q zapisuje $x_k \equiv r_k o_k$, p odczytuje $x_k \equiv r_k o_k$, po czym następuje (k+1)-ty cykl itd. Cały proces zaczyna się od pierwszego cyklu, wszystkie głowice na taśmach początkowych i roboczych są puste. Maszyny Turinga zachowujące się w ten sposób nazywamy maszynami Turinga chronologicznymi. Zanim przejdziemy dalej, odpowiednie będą pewne oznaczenia dotyczące strun.

Ciągi znaków

Ciągi znaków nad alfabetem X oznaczamy jako $s=x_1x_2...x_n$, gdzie $x_k \in X$, gdzie X jest alternatywnie interpretowane jako niepusty podzbiór IN lub samo w sobie jako bezprefiksowy zbiór ciągów znaków binarnych. Długość s wynosi $l(s)=l(x_1)+...+l(x_n)$. Analogiczne definicje obowiązują dla $y_k \in Y$. Nazywamy x_k k-tym słowem wejściowym, a y_k k-tym słowem wyjściowym (a nie literą). Ciąg znaków $s = y_1x_1...y_nx_n$ reprezentuje wejście/wyjście w kolejności chronologicznej. Ze względu na własność prefiksową x_k i y_k, s można jednoznacznie rozdzielić na jego słowa. Słowa pojawiające się w ciągach znaków są zawsze w kolejności chronologicznej. Wprowadzamy dalej następujące skróty: jest pustym ciągiem, $x_{n:m} := x_nx_{n+1}...x_{m-1}x_m$ dla n≤m i ε dla n>m. $x<n := x_1...x_{n-1}$. Analogicznie dla y. Ponadto, $yx_n := y_nx_n$, $yx_{n:m} := y_nx_n...y_mx_m$, i tak dalej.

Model AI dla znanego deterministycznego środowiska

Zdefiniujmy dla chronologicznej maszyny Turinga p funkcję częściową, również nazywaną $p : X^* \to Y^*$ z $y_{1:k} = p(x<k)$, gdzie $y_{1:k}$ jest wyjściem maszyny Turinga p na wejściu x<k w cyklu k, tj. gdzie p odczytało

do x_{k-1}, ale nie dalej.1 W analogiczny sposób definiujemy q : $Y^* \to X^*$ z $x_{1:k} = q(y_{1:k})$. Odwrotnie, dla każdej częściowej rekurencyjnej funkcji chronologicznej możemy zdefiniować odpowiadającą jej chronologiczną maszynę Turinga. Każda para (agent, środowisko) (p, q) generuje unikalną sekwencję wejścia/wyjścia $\omega^{pq} := y^{pq}_1 x^{pq}_1 y^{pq}_2 x^{pq}_2$ .... Kiedy przyjrzymy się definicjom p i q, zobaczymy ładną symetrię między systemem cybernetycznym a środowiskiem. Do tej pory nasz agent nie miał zbyt wiele inteligencji. Teraz do gry wchodzi przypisanie zasług i nieco usuwa symetrię. Dzielimy dane wejściowe $x_k \in X := R \times O$ na regularną część $o_k \in O$ i nagrodę $r_k \in R \subset IR$. Definiujemy $x_k \equiv r_k o_k$ i $r_k \equiv r(x_k)$. Celem agenta powinno być maksymalizowanie otrzymanych nagród. Nazywa się to uczeniem się przez wzmacnianie. Powodem asymetrii jest to, że ostatecznie my (ludzie) będziemy środowiskiem, z którym agent będzie się komunikował i chcemy dyktować, co jest dobre, a co złe, a nie odwrotnie. W tym jednokierunkowym uczeniu się, agent uczy się od środowiska, a nie odwrotnie, ani nie zapobiega agentowi stawaniu się bardziej inteligentnym od środowiska, ani nie zapobiega środowisku uczeniu się od agenta, ponieważ środowisko samo może interpretować wyjścia y_k jako regularną i nagrodę. Środowisko po prostu nie jest zmuszone do nauki, podczas gdy agent tak. W przypadkach, gdy ograniczamy nagrodę do dwóch wartości $r \in R = IB := \{0,1\}$, r=1 jest interpretowane jako dodatnie sprzężenie zwrotne, zwane dobrym lub poprawnym, a r=0 jako ujemne sprzężenie zwrotne, zwane złym lub błędnym. Ponadto ograniczmy na chwilę czas życia (liczbę cykli) m agenta do dużej, ale skończonej wartości. Niech

$$V_{km}^{pq} := \sum_{i=k}^{m} r(x_i^{pq})$$

niech będzie przyszłą całkowitą nagrodą (zwaną przyszłą użytecznością), którą agent p otrzymuje od środowiska q w cyklach od k do m. Teraz naturalne jest nazwanie agenta $p*$, który maksymalizuje V_{1m} (zwaną całkowitą użytecznością), najlepszym.

$$p^* := \arg\max_p V_{1m}^{pq} \quad \Rightarrow \quad V_{km}^{p^*q} \geq V_{km}^{pq} \quad \forall p : y_{<k}^{pq} = y_{<k}^{p^*q} \qquad (1)$$

Dla k=1 warunek na p jest zerowy. Dla k>1 stwierdza, że p musi być zgodne z $p*$ w tym sensie, że mają tę samą historię. Jeśli X, Y i m są skończone, liczba różnych zachowań agenta, tj. przestrzeń poszukiwań, jest skończona. Dlatego, ponieważ założyliśmy, że q jest znane, p^* można skutecznie określić, wstępnie analizując wszystkie zachowania. Głównym powodem ograniczenia do skończonego m nie było zapewnienie obliczalności $p*$, ale to, że granica $m \to \infty$ może nie istnieć. Łatwość, z jaką zdefiniowaliśmy i obliczyliśmy optymalną politykę p^*, nie jest niezwykła. Zamiast tego (nierealistyczne) założenie całkowicie znanego deterministycznego środowiska q trywializowało wszystko.

Model AI dla znanego prawdopodobieństwa a priori

Osłabmy teraz nasze założenia, zastępując deterministyczne środowisko q rozkładem prawdopodobieństwa μ(q) nad funkcjami chronologicznymi. Tutaj μ można interpretować na dwa sposoby. Albo samo środowisko zachowuje się stochastycznie zdefiniowane przez μ, albo prawdziwe środowisko jest deterministyczne, ale mamy tylko subiektywne (probabilistyczne) informacje o tym, które środowisko jest prawdziwym środowiskiem. Możliwe są również kombinacje obu przypadków. Zakładamy tutaj, że μ jest znane i opisuje prawdziwe stochastyczne zachowanie środowiska. Przypadek nieznanego μ z agentem mającym pewne przekonania na temat środowiska leży u podstaw modelu AIξ. Najlepszym lub najinteligentniejszym agentem jest teraz ten, który maksymalizuje oczekiwaną użyteczność (nazywaną funkcją wartości) $V^p_\mu \equiv V^{p\mu}_{1m} := \Sigma_q \mu(q) V^{pq}_{1m}$. To definiuje model AIμ.

Definicja 2 (model AIμ). Model AIμ to agent z polityką p^μ, która maksymalizuje μ-oczekiwaną całkowitą nagrodę $r_1+...+r_m$, tj. $p^* \equiv p^\mu := \text{argmax}_p V^p_\mu$. Jego wartość to $V^*_\mu := V^{p\mu}_\mu$.

Potrzebujemy koncepcji funkcji wartości w nieco bardziej ogólnej formie.

Definicja 3 (funkcja μ/prawda/generująca wartość). Percepcja agenta x składa się z regularnej obserwacji o∈O i nagrody r∈R⊂IR. W cyklu k wartość $V^{p\mu}_{km}(yx_{<k})$ jest definiowana jako μ-oczekiwanie przyszłej sumy nagród $r_k+...+r_m$ z działaniami generowanymi przez politykę p i ustaloną historią $yx_{<k}$. Mówimy, że $V^{p\mu}_{km}(yx_{<k})$ jest (przyszłą) wartością polityki p w środowisku μ przy danej historii $yx_{<k}$ lub krócej, μ lub prawdziwą lub generującą wartością p przy danej $yx_{<k}$. $V^p_\mu := V^{p\mu}_{1m}$ jest (całkowitą) wartością p.

Podajemy teraz bardziej formalną definicję dla $V^{p\mu}_{km}$. Załóżmy, że jesteśmy w cyklu k z historią $\dot{y}\dot{x}_1...\dot{y}\dot{x}_{k-1}$ i poprośmy o najlepszy wynik y_k. Ponadto niech $\dot{Q}_k := \{q : q(\dot{y}_{<k}) = \dot{x}_{<k}\}$ będzie zbiorem wszystkich środowisk produkujących powyższą historię. Mówimy, że $q \in \dot{Q}_k$ jest zgodne z historią $\dot{y}\dot{x}_{<k}$. Oczekiwana nagroda za następne m−k+1 cykli (biorąc pod uwagę powyższą historię) nazywana jest wartością polityki p i jest dana przez prawdopodobieństwo warunkowe:

$$V^{p\mu}_{km}(\dot{y}\dot{x}_{<k}) := \frac{\sum_{q\in\dot{Q}_k}\mu(q)V^{pq}_{km}}{\sum_{q\in\dot{Q}_k}\mu(q)}. \qquad (2)$$

Polityka p i środowisko μ nie determinują historii $\dot{y}\dot{x}_{<k}$, w przeciwieństwie do przypadku deterministycznego, ponieważ historia nie jest już deterministycznie determinowana przez p i q, ale zależy od p i μ oraz od wyniku procesu stochastycznego. Każdy nowy cykl dodaje nową informację $(\dot{x}_i)$ do agenta. Jest to wskazane przez kropki nad symbolami. W cyklu k musimy zmaksymalizować oczekiwane przyszłe nagrody, biorąc pod uwagę informacje w historii $\dot{y}\dot{x}_{<k}$. Informacje te nie są już obecne w p i q/μ na początku agenta, w przeciwieństwie do przypadku deterministycznego. Ponadto chcemy uogólnić skończony czas życia m na dynamiczną (obliczalną) dalekowzroczność $h_k \equiv m_k - k + 1 \geq 1$, zwaną horyzontem. Dla $m_k=m$ mamy nasz pierwotny skończony czas życia; dla $h_k=h$ agent maksymalizuje w każdym cyklu następne h oczekiwanych nagród. Następne nagrody h_k są maksymalizowane przez

$$p^*_k := \arg\max_{p\in\dot{P}_k} V^{p\mu}_{km_k}(\dot{y}\dot{x}_{<k})$$

gdzie $\dot{P}_k := \{p : \exists y_k : p(\dot{x}_{<k}) = \dot{y}_{<k}y_k\}$ jest zbiorem systemów zgodnych z bieżącą historią. Należy zauważyć, że p^*_k zależy od k i jest używane tylko w kroku k do określenia $\dot{y}_k$ przez $p^*_k(\dot{x}_{<k}|\dot{y}_{<k}) = \dot{y}_{<k}\dot{y}_k$. Po zapisaniu $\dot{y}_k$ środowisko odpowiada $\dot{x}_k$ z (warunkowym) prawdopodobieństwem $\mu(\dot{Q}_{k+1})/\mu(\dot{Q}_k)$. Ten prawdopodobny wynik dostarcza agentowi nowych informacji. Cykl k+1 rozpoczyna się od określenia $\dot{y}_{k+1}$ z p^*_{k+1} (które może różnić się od p^*_k dla

dynamicznego m_k) i tak dalej. Należy zauważyć, że p^*_k niejawnie zależy również od $\dot{y}_{<k}$, ponieważ $\dot{P}_k$ i $\dot{Q}_k$ tak robią. Ale rekurencyjnie wstawiając p^*_{k-1} i tak dalej, możemy zdefiniować

$$p^*(\dot{x}_{<k}) \;:=\; p^*_k(\dot{x}_{<k}|p^*_{k-1}(\dot{x}_{<k-1}|...p^*_1)). \qquad (3)$$

Jest to funkcja chronologiczna i obliczalna, jeśli X, Y i m_k są skończone, a μ jest obliczalny. Dla stałej m można pokazać, że polityka (3) pokrywa się z modelem AIμ (definicja 2). To również dowodzi

$$V^{*\mu}_{km}(yx_{<k}) \geq V^{p\mu}_{km}(yx_{<k}) \quad \forall p \text{ consistent with } yx_{<k}, \qquad (4)$$

podobnie do (1). Dla k = 1 jest to oczywiste. Nazywamy również (3) modelem AIμ. Dla deterministycznego μ ten model redukuje się do przypadku deterministycznego omawianego w ostatniej podsekcji. Ważne jest, aby maksymalizować sumę przyszłych nagród, a nie na przykład być zachłannym i maksymalizować tylko następną nagrodę, jak to się robi np. w przewidywaniu sekwencji. Na przykład niech otoczenie będzie sekwencją gier szachowych, a każdy cykl odpowiada jednemu ruchowi. Tylko na końcu każdej gry agent otrzymuje pozytywną nagrodę r = 1, jeśli wygrał grę (i nie wykonał żadnego nielegalnego ruchu). Dla agenta maksymalizacja wszystkich przyszłych nagród oznacza próbę wygrania jak największej liczby gier w jak najkrótszym czasie (i unikanie nielegalnych ruchów). Tę samą wydajność osiągamy, jeśli wybierzemy h_k znacznie większe niż typowe długości gier. Maksymalizacja tylko następnej nagrody byłaby bardzo złym agentem grającym w szachy. Nawet gdybyśmy uczynili naszą nagrodę r drobniejszą, np. oceniając liczbę figur szachowych, agent grałby bardzo źle w szachy dla h_k=1, rzeczywiście. Model AIμ nadal zależy od μ i m_k. Aby uzyskać nasz ostateczny uniwersalny model AI, pomysł polega na zastąpieniu μ uniwersalnym prawdopodobieństwem ξ, zdefiniowanym później. Jest to motywowane faktem, że ξ zbiega się do μ w pewnym sensie dla dowolnego μ. Przy ξ zamiast μ nasz model nie zależy już od żadnych parametrów, więc jest prawdziwie uniwersalny. Pozostaje pokazać, że zachowuje się inteligentnie. Ale kontynuujmy krok po kroku. W dalszej części opracowujemy alternatywną, ale równoważną formułę modelu AIμ. Podczas gdy forma funkcjonalna przedstawiona powyżej jest bardziej odpowiednia do rozważań teoretycznych, iteratywna i rekurencyjna formuła z następnych podsekcji będzie bardziej odpowiednia do jawnych obliczeń w większości innych sekcji.

Rozkłady prawdopodobieństwa

Używamy liter greckich do rozkładów prawdopodobieństwa i podkreślamy ich argumenty, aby wskazać, że są to argumenty prawdopodobieństwa. Niech $\rho_n(x_1...x_n)$ będzie prawdopodobieństwem, że (nieskończony) ciąg zaczyna się od $x_1...x_n$. Usuwamy indeks na ρ, jeśli wynika to jasno z jego argumentów:

$$\sum_{x_n\in\mathcal{X}} \rho(\underline{x}_{1:n}) \equiv \sum_{x_n} \rho_n(\underline{x}_{1:n}) = \rho_{n-1}(\underline{x}_{<n}) \equiv \rho(\underline{x}_{<n}), \quad \rho(\epsilon) \equiv \rho_0(\epsilon) = 1. \quad (5)$$

Potrzebujemy również prawdopodobieństw warunkowych wyprowadzonych z reguły łańcuchowej. Preferujemy notację, która zachowuje chronologiczną kolejność słów, w przeciwieństwie do standardowej notacji $\rho(\cdot|\cdot)$, która ją odwraca. Rozszerzamy definicję ρ na przypadek warunkowy, stosując następującą konwencję dla jego argumentów: Podkreślony argument x_k jest zmienną prawdopodobieństwa, a inne niepodkreślone argumenty x_k reprezentują warunki. Zgodnie z tą konwencją prawdopodobieństwo warunkowe ma postać $\rho(x_{<n}\underline{x}_n) = \rho(\underline{x}_{1:n})/\rho(\underline{x}_{<n})$. Równanie

stwierdza, że prawdopodobieństwo, że po ciągu x1...x_{n-1} następuje x_n, jest równe prawdopodobieństwu $x_1...x_{n-1}*$ podzielonemu przez prawdopodobieństwo $x_1...x_{n-1}*$. Używamy $x*$ jako skrótu dla „ciągów zaczynających się od x". Wprowadzona notacja nadaje się również do definiowania prawdopodobieństwa warunkowego $\rho(y_1\underline{x}_1...y_n\underline{x}_n)$, że środowisko reaguje z $x_1...x_n$ pod warunkiem, że wyjście agenta to $y_1...y_n$. Środowisko jest chronologiczne, tj. wejście x_i zależy tylko od $yx<_i y_i$. W przypadku probabilistycznym oznacza to, że $\rho(y\underline{x}_{<k}y_k) := \sum_{x_k}\rho(y\underline{x}_{1:k})$ jest niezależne od y_k, stąd ogon y_k w argumentach ρ może zostać pominięty. Rozkłady prawdopodobieństwa z tą własnością będą nazywane chronologicznymi. Y są zawsze warunkami, tj. nigdy nie są podkreślone, podczas gdy dodatkowe warunkowanie dla x można uzyskać za pomocą reguły łańcuchowej:

$$\rho(yx_{<n}y\underline{x}_n) = \rho(y\underline{x}_{1:n})/\rho(y\underline{x}_{<n}) \quad \text{and} \qquad (6)$$
$$\rho(y\underline{x}_{1:n}) = \rho(y\underline{x}_1)\cdot\rho(yx_1y\underline{x}_2)\cdot...\cdot\rho(yx_{<n}y\underline{x}_n).$$

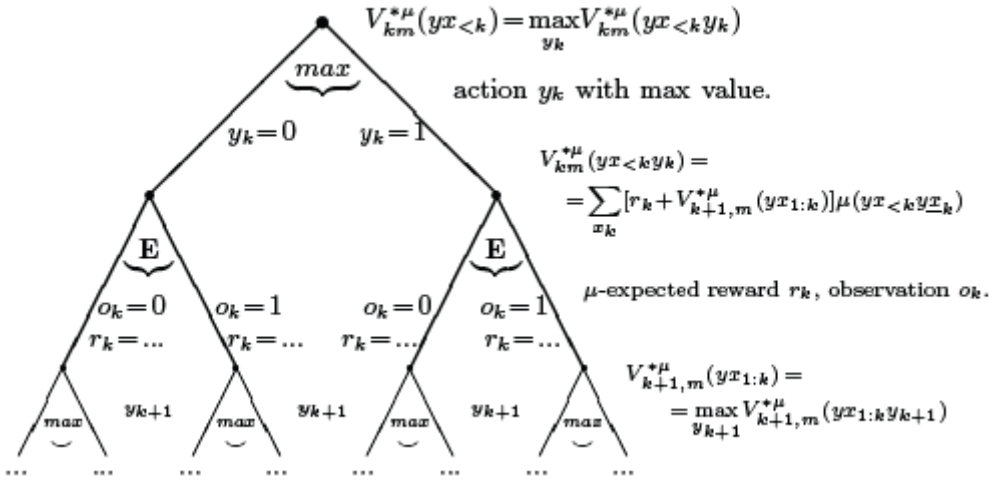
Drugie równanie jest pierwszym równaniem zastosowanym n razy.

Jawna forma modelu AIμ

Zdefiniujmy model AIμ p^* w inny sposób: Niech $\mu(yx_{<k}y\underline{x}_k)$ będzie prawdziwym prawdopodobieństwem wejścia x_k w cyklu k, biorąc pod uwagę historię $yx_{<k}y_k$; $\mu(y\underline{x}_{1:k})$ jest prawdziwym chronologicznym prawdopodobieństwem wcześniejszym, że środowisko reaguje z $x_{1:k}$, jeśli agent dostarczy mu działania $y_{1:k}$. Zakładamy, że model cybernetyczny jest poprawny. Następnie definiujemy wartość $V^{*\mu}_{k+1,m}(yx_{1:k})$ jako μ-oczekiwaną sumę nagród $r_{k+1}+...+r_m$ w cyklach k+1 do m z wyjściami y_i generowanymi przez agenta p^*, która maksymalizuje oczekiwaną sumę nagród i odpowiedzi xi ze środowiska, wylosowane zgodnie z μ. Dodając $r(x_k)\equiv r_k$ otrzymujemy nagrodę obejmującą cykl k. Prawdopodobieństwo x_k, biorąc pod uwagę $yx<_k y_k$, jest podane przez prawdopodobieństwo warunkowe $\mu(yx_{<k}y\underline{x}_k)$. Tak więc oczekiwana suma nagród w cyklach od k do m, biorąc pod uwagę $yx<_k y_k$, wynosi

$$V^{*\mu}_{km}(yx_{<k}y_k) := \sum_{x_k}[r(x_k)+V^{*\mu}_{k+1,m}(yx_{1:k})]\cdot\mu(yx_{<k}y\underline{x}_k). \qquad (7)$$

Teraz pytamy, jak p^* wybiera y_k: Powinien wybrać y_k, aby zmaksymalizować przyszłe nagrody. Tak więc oczekiwana nagroda w cyklach od k do m, biorąc pod uwagę $yx_{<k}$ i y_k wybrane przez p^*, wynosi $V^{*\mu}_{km}(yx_{<k}) := \max_{y_k} V^{*\mu}_{km}(yx<_k y_k)$.



Wraz z początkiem indukcji

$$V^{*\mu}_{m+1,m}(yx_{1:m}) \;:=\; 0, \qquad (8)$$

$V^{*\mu}_{km}$ jest całkowicie zdefiniowane. Możemy podsumować jeden cykl wzorem

$$V^{*\mu}_{km}(yx_{<k}) \;=\; \max_{y_k}\sum_{x_k}[r(x_k)+V^{*\mu}_{k+1,m}(yx_{1:k})]\cdot\mu(yx_{<k}y\underline{x}_k). \qquad (9)$$

Wprowadzamy dynamiczną (obliczalną) dalekowzroczność $h_k \equiv m_k - k + 1 \geq 1$, zwaną horyzontem. Dla $m_k=m$, gdzie m jest czasem życia agenta, osiągamy optymalne zachowanie, dla ograniczonej dalekowzroczności $h_k = h$ ($m = mk = h+k-1$), agent maksymalizuje w każdym cyklu następne h oczekiwanych nagród. Jeśli m_k jest naszą funkcją horyzontu p^*, a $\dot{y}\dot{x}_{<k}$ jest rzeczywistą historią w cyklu k, wyjście $\dot{y}_k$ agenta jest jawnie podane przez

$$\dot{y}_k \;=\; \arg\max_{y_k} V^{*\mu}_{km_k}(\dot{y}\dot{x}_{<k}y_k), \qquad (10)$$

co z kolei definiuje politykę p^*. Następnie środowisko odpowiada $\dot{x}_k$ z prawdopodobieństwem $\mu(\dot{y}\dot{x}_{<k}\dot{y}\underline{\dot{x}}_k)$. Następnie rozpoczyna się cykl k+1. Możemy rozwinąć rekurencję (9) dalej i podać $\dot{y}_k$ nierekurencyjnie jako

$$\dot{y}_k \equiv \dot{y}^{\mu}_k := \arg\max_{y_k}\sum_{x_k}\max_{y_{k+1}}\sum_{x_{k+1}}...\max_{y_{m_k}}\sum_{x_{m_k}}(r(x_k)+...+r(x_{m_k}))\cdot\mu(\dot{y}\dot{x}_{<k}y\underline{x}_{k:m_k}). \qquad (11)$$

Ma to bezpośrednią interpretację: prawdopodobieństwo wejść $x_{k:mk}$ w cyklu k, gdy agent wyprowadza $y_{k:mk}$ z rzeczywistą historią $\dot{y}\dot{x}_{<k}$ is $\mu(\dot{y}\dot{x}_{<k}y\underline{x}_{k:m_k})$. Przyszła nagroda w tym przypadku wynosi $r(x_k)+...+r(x_{mk})$. Najlepszą oczekiwaną nagrodę uzyskuje się przez uśrednienie w x_i (Σx_i) i maksymalizację w y_i. Należy to zrobić w kolejności chronologicznej, aby poprawnie uwzględnić zależności x_i i y_i w historii. Jest to zasadniczo algorytm/drzewo expectimax. Model AIμ jest optymalny w tym sensie, że żadna inna polityka nie prowadzi do wyższej oczekiwanej nagrody. Wartość ogólnej polityki p można zapisać w postaci

$$V_{km}^{p\mu}(yx_{<k}) \;:=\; \sum_{x_{1:m}} (r_k + ... + r_m)\mu(yx_{<k}y\underline{x}_{k:m})_{|y_{1:m}=p(x_{<m})}. \qquad (12)$$

Jak wynika z ich interpretacji, iteracyjne prawdopodobieństwo środowiskowe μ wiąże się z formą funkcyjną w następujący sposób:

$$\mu(y\underline{x}_{1:k}) \;=\; \sum_{q:q(y_{1:k})=x_{1:k}} \mu(q) \qquad (13)$$

Dzięki tej identyfikacji można wykazać co następuje:

Twierdzenie 2 (Równoważność funkcjonalnego i jawnego modelu AI).

Działania funkcjonalnego modelu AI (3) pokrywają się z działaniami jawnego (rekurencyjnego/iteracyjnego) modelu AI (9)–(11) ze środowiskami zidentyfikowanymi przez (13).

Środowiska czynnikowe

Do tej pory nie nałożyliśmy żadnych ograniczeń na formę prawdopodobieństwa a priori μ, poza tym, że jest to rozkład prawdopodobieństwa chronologicznego. Z drugiej strony zobaczymy, że aby udowodnić ścisłe ograniczenia nagrody, prawdopodobieństwo a priori musi spełniać pewien warunek rozdzielności, który zostanie zdefiniowany później. Tutaj wprowadzamy bardzo silną formę rozdzielności, gdy μ rozkłada się na iloczyny. Załóżmy, że cykle są pogrupowane w niezależne odcinki r=1,2,3,..., gdzie każdy odcinek r składa się z cykli $k=n_r+1,...,n_{r+1}$ dla pewnego $0=n_0< n1<...<n_s=n$:

$$\mu(y\underline{x}_{1:n}) \;=\; \prod_{r=0}^{s-1} \mu_r(y\underline{x}_{n_r+1:n_{r+1}}) \qquad (14)$$

(W najprostszym przypadku, gdy wszystkie odcinki mają taką samą długość l, wówczas $n_r=r\cdot l$). Wówczas $\dot{y}_k$ zależy tylko od μ_r oraz x i y odcinka r, przy czym r jest takie, że $n_r<k\leq n_{r+1}$. Można pokazać, że

$$\dot{y}_k \;=\; \arg\max_{y_k} V_{km_k}^{*\mu}(\dot{y}\dot{x}_{<k}y_k) \;=\; \arg\max_{y_k} V_{kt}^{*\mu}(\dot{y}\dot{x}_{<k}y_k), \qquad (15)$$

z $t:=\min\{m_k,n_{r+1}\}$. Różne odcinki są całkowicie niezależne w tym sensie, że wejścia x_k różnych odcinków są statystycznie niezależne i zależą tylko od wyjść y_k tego samego odcinka. Wyjścia y_k zależą tylko od x i y odpowiadającego odcinka r i są niezależne od faktycznego wejścia/wyjścia innych odcinków. Należy zauważyć, że $\dot{y}_k$ jest również niezależne od wyboru m_k, o ile m_k jest wystarczająco duże. Jeśli wszystkie odcinki mają długość co najwyżej l, tj. $n_{r+1}-n_r\leq l$ i jeśli wybierzemy horyzont h_k wynoszący co najmniej l, to $m_k\geq k+l-1\geq n_r+l\geq n_{r+1}$ i stąd $t = n_{r+1}$ niezależnie od m_k. Oznacza to, że dla czynnikowalnego μ nie ma problemu z przyjęciem granicy $m_k\to\infty$. Być może to ograniczenie można również wykonać w bardziej ogólnym przypadku wystarczająco rozdzielnego μ. Problem wyboru m_k zostanie omówiony bardziej szczegółowo później. Chociaż faktoryzowalne μ są zbyt restrykcyjne, aby objąć wszystkie problemy AI, często występują w praktyce w formie powtarzanego rozwiązywania problemów, a zatem są warte zbadania. Na przykład, jeśli agent musi wielokrotnie grać w gry takie jak szachy lub musi minimalizować różne funkcje, różne gry/funkcje mogą być całkowicie niezależne, tj. prawdopodobieństwo środowiskowe ulega czynnikowaniu, gdzie każdy czynnik odpowiada minimalizacji gry/funkcji. Aby uzyskać szczegółowe informacje, zobacz odpowiednie sekcje dotyczące gier strategicznych i minimalizacji funkcji. Ponadto, dla faktoryzowalnego μ prawdopodobnie łatwiej jest wyprowadzić

odpowiednie granice nagrody dla uniwersalnego modelu AIξ zdefiniowanego w następnej sekcji, niż dla przypadków rozdzielnych, które zostaną wprowadzone później. Może to być pierwszy krok w kierunku definicji i dowodu dla ogólnego przypadku problemów rozdzielnych. Celem tego akapitu było pokazanie, że pojęcie faktoryzowalnego μ może być pierwszym krokiem w kierunku definicji i analizy ogólnego przypadku rozdzielnego μ.

Stałe i ograniczenia

Mamy na myśli uniwersalnego agenta ze złożonymi interakcjami, który jest co najmniej tak inteligentny i złożony jak człowiek. Można pomyśleć o agencie, którego dane wejściowe y_k pochodzą z cyfrowej kamery wideo, a dane wyjściowe x_k to jakiś obraz na monitorze4, tylko dla nagród, które moglibyśmy ograniczyć do najbardziej prymitywnych binarnych, tj. $r_k \in IB$. Tak więc myślimy o następujących stałych rozmiarach:

$$\begin{array}{ccccccc} 1 & \ll & \langle l(y_k x_k) \rangle & \ll & k \leq m & \ll & |\mathcal{Y} \times \mathcal{X}| \\ 1 & \ll & 2^{16} & \ll & 2^{24} \leq 2^{32} & \ll & 2^{65536} \end{array}$$

Pierwsze dwa ograniczenia mówią, że rzeczywista liczba k wejść/wyjść powinna być rozsądnie duża w porównaniu do typowej długości l słów wejścia/wyjścia, która sama w sobie powinna być dość spora. Ostatnie ograniczenie wyraża fakt, że całkowity czas życia m (liczba cykli wejścia/wyjścia) agenta jest o wiele za krótki, aby umożliwić wystąpienie każdego możliwego wejścia, lub wypróbować każde możliwe wyjście, lub wykorzystać identycznie powtarzane wejścia lub wyjścia. Nie spodziewamy się żadnych użytecznych wyjść dla $k \lesssim \langle l \rangle$. Bardziej interesująca niż długości wejść jest złożoność $K(x_1...x_k)$ wszystkich wejść do tej pory, która zostanie zdefiniowana później. Środowisko zwykle nie jest „doskonałe". Agent może albo oddziaływać z niedoskonałym człowiekiem, albo zmierzyć się ze światem niedeterministycznym (z powodu mechaniki kwantowej lub chaosu).5 W obu przypadkach sekwencja zawiera pewien szum, co prowadzi do $K(x_1...x_k) \propto \langle l \rangle \cdot k$. Złożoność rozkładu prawdopodobieństwa sekwencji wejściowej jest czymś innym. Zakładamy, że ten hałaśliwy świat działa zgodnie z kilkoma prostymi, obliczalnymi zasadami. $K(\mu_k) \ll \langle l \rangle \cdot k$, tj. zasady świata mogą być wysoce skompresowane. Możemy dopuścić środowiska, w których pojawiają się nowe aspekty dla $k \to \infty$, powodując nieograniczony $K(\mu_k)$.

W dalszej części nigdy nie używamy tych ograniczeń, chyba że jest to wyraźnie określone. W niektórych prostszych modelach i przykładach rozmiar stałych będzie nawet naruszał te ograniczenia (np. $l(x_k)=l(y_k)=1$), ale to właśnie powyższe ograniczenia powinien mieć na uwadze czytelnik. Interesują nas tylko twierdzenia, które nie degenerują się w ramach powyższych ograniczeń. Aby uniknąć uciążliwych rozważań dotyczących zbieżności i istnienia, w całej pracy przyjmujemy następujące założenia:

Założenie 3 (Skończoność) Zakładamy, że:

• przestrzeń wejścia/percepcji X jest skończona

• przestrzeń wyjścia/działania Y jest skończona

• nagrody są nieujemne i ograniczone, tj. $r_k \in R \subseteq [0, r_{max}]$,

• horyzont m jest skończony

Skończony X i ograniczony R (każdy osobno) zapewniają istnienie μ-oczekiwań, ale czasami są potrzebne razem. Skończony Y zapewnia, że $\arg\max_{y_k \in Y}[...]$ istnieje, tj. że osiągane są maksima, podczas gdy skończony m unika różnych problemów technicznych i filozoficznych, a pozytywne

nagrody są potrzebne dla ograniczonego czasowo modelu AIXItl. Wiele twierdzeń można uogólnić, rozluźniając niektóre lub wszystkie z powyższych założeń skończoności.

Sekwencyjna teoria decyzji

Można powiązać (9) z równaniami Bellmana sekwencyjnej teorii decyzji, identyfikując kompletne historie $yx_{<k}$ ze stanami, $\mu(yx_{<k}y\underline{x}_k)$ z macierzą przejść stanów, V^*_μ z funkcją wartości i y_k z działaniem w cyklu k. Ze względu na użycie kompletnych historii jako przestrzeni stanów, model AIμ nie zakłada ani stacjonarności, ani własności Markowa, ani całkowitej dostępności środowiska. Każdy stan występuje co najwyżej raz w okresie życia systemu. Z tego i innych powodów jawne sformułowanie jest tutaj bardziej naturalne i użyteczne niż wymuszanie pseudorekurencyjnej formy równania Bellmana. Ponieważ mamy na myśli uniwersalny system ze złożonymi interakcjami, przestrzenie działania i percepcji Y i X są ogromne (np. obrazy wideo), a każde działanie lub samo postrzeganie występuje zwykle tylko raz w okresie życia m agenta. Ponieważ nie istnieje (oczywista) uniwersalna relacja podobieństwa w przestrzeni stanów, skuteczna redukcja jej rozmiaru jest niemożliwa, ale nie ma zasadniczego problemu w określeniu y_k , o ile μ jest znane i obliczalne, a X, Y i m są skończone. Rzeczy drastycznie się zmieniają, jeśli μ jest nieznane. Algorytmy uczenia się przez wzmacnianie są powszechnie używane w tym przypadku do nauki nieznanego μ lub bezpośrednio jego wartości. Udaje im się to, jeśli przestrzeń stanów jest mała lub została skutecznie zmniejszona przez techniki generalizacji lub aproksymacji funkcji. W każdym przypadku rozwiązania są albo ad hoc, działają tylko w ograniczonych domenach, mają poważne problemy z eksploracją przestrzeni stanów w porównaniu z eksploracją, lub są podatne na rozbieżności lub mają nieoptymalne wskaźniki uczenia się. Jak dotąd nie ma uniwersalnego i optymalnego rozwiązania tego problemu. Głównym tematem tego artykułu jest przedstawienie nowego modelu i argumentowanie, że formalnie rozwiązuje on wszystkie te problemy w optymalny sposób. Prawdziwy rozkład prawdopodobieństwa μ nie zostanie poznany bezpośrednio, ale zostanie zastąpiony przez pewien uogólniony uniwersalny rozkład a priori ξ, który zbiega się do μ.

Uniwersalna predykcja sekwencji

Ta sekcja zajmuje się kwestią, jak dokonywać przewidywań w nieznanym otoczeniu. Po krótkim opisie ważnych postaw filozoficznych dotyczących rozumowania indukcyjnego i wnioskowania, dokładniej opisujemy, co rozumiemy przez indukcję, i wyjaśniamy, dlaczego możemy skupić się na zadaniach przewidywania sekwencji. Najważniejszą koncepcją jest zasada brzytwy Ockhama (prostota). Rzeczywiście, można wykazać, że najlepszym sposobem dokonywania przewidywań jest oparcie się na najkrótszym (= najprostrzym) opisie sekwencji danych, jaki do tej pory widziano. Najbardziej ogólne skuteczne opisy można uzyskać za pomocą ogólnych funkcji rekurencyjnych lub równoważnie, używając programów na maszynach Turinga, zwłaszcza na uniwersalnej maszynie Turinga. Długość najkrótszego programu opisującego dane nazywa się złożonością danych Kołmogorowa. Teoria prawdopodobieństwa jest potrzebna do radzenia sobie z niepewnością. Otoczenie może być procesem stochastycznym (np. domy gry lub fizyka kwantowa), który można opisać za pomocą „obiektywnych" prawdopodobieństw. Ale także niepewna wiedza o środowisku, która prowadzi do przekonań na jego temat, może być modelowana przez „subiektywne" prawdopodobieństwa. Stare pytanie pozostawione otwarte przez subiektywistów, jak wybierać a priori prawdopodobieństwa, jest rozwiązywane przez uniwersalne prawdopodobieństwo a priori Solomonoffa, które jest ściśle powiązane ze złożonością Kolmogorova. Głównym wynikiem Solomonoffa jest to, że uniwersalne (subiektywne) a posteriori zbiega się do prawdziwego (obiektywnego) środowiska (al prawdopodobieństwa) μ. Jedynym założeniem dotyczącym μ jest to, że μ (które nie musi być znane!)

jest obliczalne. Problem nieznanego środowiska μ jest zatem rozwiązany dla wszystkich problemów typu indukcyjnego, takich jak przewidywanie sekwencji i klasyfikacja.

Wprowadzenie

Ważnym i wysoce nietrywialnym aspektem inteligencji jest wnioskowanie indukcyjne. Mówiąc prościej, indukcja to proces przewidywania przyszłości na podstawie przeszłości, a dokładniej, proces znajdowania reguł w (przeszłych) danych i wykorzystywania tych reguł do odgadywania przyszłych danych. Nietrywialnymi przykładami są prognozowanie pogody lub rynku akcji lub ciągłe serie liczbowe w teście IQ. Tworzenie dobrych prognoz odgrywa centralną rolę w inteligencji naturalnej i sztucznej w ogóle, a w szczególności w uczeniu maszynowym. Wszystkie problemy indukcyjne można sformułować jako zadania przewidywania sekwencji. Jest to na przykład oczywiste w przypadku przewidywania szeregów czasowych, ale obejmuje również zadania klasyfikacyjne. Po zaobserwowaniu danych x_t w czasach t<n, zadaniem jest przewidzenie n-tego symbolu x_n z sekwencji $x_1...x_{n-1}$. To podejście presekwencyjne pomija pośredni krok uczenia się modelu na podstawie zaobserwowanych danych $x_1...x_{n-1}$, a następnie wykorzystania tego modelu do przewidywania x_n. Podejście preqencjialne unika problemów spójności modelu, sposobu oddzielania szumu od użytecznych danych i wielu innych kwestii. Celem jest dokonywanie „dobrych" przewidywań, gdzie jakość przewidywań jest zwykle mierzona funkcją straty, która musi zostać zminimalizowana. Kluczową koncepcją dobrego definiowania i rozwiązywania problemów indukcyjnych jest zasada brzytwy Ockhama (prostota), która mówi, że „Bytów nie należy mnożyć ponad konieczność", co można interpretować jako zachowanie najprostszej teorii zgodnej z obserwacjami $x_1...x_{n-1}$ i wykorzystanie tej teorii do przewidywania x_n. Zanim będziemy mogli przedstawić formalne rozwiązanie Solomonoffa, musimy skwantyfikować brzytwę Ockhama w kategoriach złożoności Kolmogorowa i wprowadzić pojęcie prawdopodobieństw subiektywnych/obiektywnych.

Algorytmiczna teoria informacji

Intuicyjnie, ciąg jest prosty, jeśli można go opisać kilkoma słowami, jak „ciąg miliona jedynek", i jest złożony, jeśli nie ma takiego krótkiego opisu, jak w przypadku losowego ciągu, którego najkrótszym opisem jest określenie go bit po bicie. Możemy ograniczyć dyskusję do ciągów binarnych, ponieważ dla innych (nieciągowych) obiektów matematycznych możemy założyć pewne domyślne kodowanie jako ciągi binarne. Ponadto interesują nas tylko opisy efektywne, a zatem ograniczamy dekodery do maszyn Turinga. Wybierzmy jakąś uniwersalną (tzw. prefiksową) maszynę Turinga U z jednokierunkowymi taśmami wejściowymi i wyjściowymi binarnymi oraz dwukierunkową taśmą roboczą. Następnie możemy zdefiniować (warunkową) złożoność prefiksową Kolmogorowa ciągu binarnego x jako długość l najkrótszego programu p, dla którego U wyprowadza ciąg binarny x (przy danym y).

Definicja 4 (złożoność Kolmogorowa). Niech U będzie uniwersalną maszyną Turinga prefiksową U. (Warunkowa) złożoność prefiksowa Kolmogorowa jest zdefiniowana jako najkrótszy program p, dla którego U wyprowadza x (przy danym y):

$$K(x) := \min_p\{l(p) : U(p) = x\}, \quad K(x|y) := \min_p\{l(p) : U(y,p) = x\}$$

Proste ciągi znaków, takie jak 000...0, mogą być generowane przez krótkie programy, a zatem mają niską złożoność Kolmogorowa, ale nieregularne (np. losowe) ciągi znaków są ich najkrótszym opisem, a zatem mają wysoką złożoność Kolmogorowa. Ważną właściwością K jest to, że jest niemal niezależna od wyboru U. Ponadto dzieli wiele właściwości z entropią Shannona (miarą informacji) S, ale K jest lepsza od S pod wieloma względami. Krótko mówiąc, K jest doskonałą uniwersalną miarą złożoności, nadającą się do kwantyfikacji brzytwy Ockhama. Istnieje (tylko) jedna poważna wada: K nie jest

skończenie obliczalne. Główną właściwością algorytmiczną K jest to, że jest (tylko) współprzeliczalna, tj. jest aproksymowalna z góry. Dla ogólnych obiektów (nieciągów) można określić pewne domyślne kodowanie · i zdefiniować K(obiekt) :=K(obiekt), szczególnie dla liczb i par, np. skracamy K(x,y):=K(x,y). Najważniejsze własności informacyjno-teoretyczne K są wymienione poniżej, gdzie skracamy f(x)≤g(x)+O(1) przez $f(x) \stackrel{+}{\leq} g(x)$. Później skracamy również f(x)=O(g(x)) przez $f(x) \stackrel{\times}{\leq} g(x)$.

Twierdzenie 4 (Własności informacyjne złożoności Kołmogorowa).

$$
\begin{aligned}
&i)\ \ K(x) \stackrel{+}{\leq} l(x)+2\log l(x), \qquad K(n) \stackrel{+}{\leq} \log n+2\log\log n.\\
&ii)\ \textstyle\sum_x 2^{-K(x)} \leq 1, \quad K(x) \geq l(x) \text{ for 'most' } x, \quad K(n)\to\infty \text{ for } n\to\infty.\\
&iii) K(x|y) \stackrel{+}{\leq} K(x) \stackrel{+}{\leq} K(x,y).\\
&iv)\ K(x,y) \stackrel{+}{\leq} K(x)+K(y), \qquad K(xy) \stackrel{+}{\leq} K(x)+K(y).\\
&v)\ \ K(x|y,K(y))+K(y) \stackrel{+}{=} K(x,y) \stackrel{+}{=} K(y,x) \stackrel{+}{=} K(y|x,K(x))+K(x).\\
&vi)\ K(f(x)) \stackrel{+}{\leq} K(x)+K(f) \ \text{ if } \ f:IB^*\to IB^* \text{ is recursive/computable.}\\
&vii) K(x) \stackrel{+}{\leq} -\log_2 P(x)+K(P) \text{ if } P:IB^*\to[0,1] \text{ is recursive and } \textstyle\sum_x P(x)\leq 1
\end{aligned}
$$

Wszystkie (nie)równości pozostają ważne, jeśli K jest (dalej) warunkowane przy pewnym z, tj. $K(...)\rightsquigarrow K(...|z)$ i $K(...|y)\rightsquigarrow K(...|y,z)$. Wszystkie podane są ważne w ramach stałej addytywnej o rozmiarze O(1), ale są inne, które są ważne tylko do dokładności logarytmicznej. K ma wiele wspólnych właściwości z entropią Shannona, jak być powinno, ponieważ obie mierzą zawartość informacyjną ciągu. Własność (i) podaje górną granicę K, a własność (ii) jest nierównością Krafta, która implikuje dolną granicę K ważną dla „większości" n, gdzie „większość" oznacza, że istnieją tylko o(N) wyjątków dla n∈{1,...,N}. Podanie informacji pobocznych y nigdy nie może zwiększyć długości kodu, a wymaganie dodatkowych informacji y nigdy nie może zmniejszyć długości kodu (iii). Kodowanie x i y oddzielnie nigdy nie pomaga (iv), a przekształcenie x nie zwiększa jego zawartości informacyjnej (vi). Własność (vi) pokazuje również, że jeśli x koduje jakiś obiekt o, przełączanie się z jednego schematu kodowania na inny za pomocą rekurencyjnej bijekcji pozostawia K niezmienione w obrębie addytywnych terminów O(1). Pierwszym nietrywialnym wynikiem jest symetria informacji (v), która jest odpowiednikiem reguły mnożenia/łańcucha dla prawdopodobieństw warunkowych. Własność (vii) leży u podstaw zasady MDL [52], która aproksymuje K(x) przez $-\log_2 P(x)+K(P)$.

Niepewność i prawdopodobieństwa

Dla obiektywisty prawdopodobieństwa są rzeczywistymi aspektami świata. Wynik obserwacji lub eksperymentu nie jest deterministyczny, ale obejmuje fizyczne procesy losowe. Aksjomaty teorii prawdopodobieństwa Kołmogorowa formalizują właściwości, jakie powinny mieć prawdopodobieństwa. W przypadku eksperymentów i.i.d. prawdopodobieństwa przypisane zdarzeniom można interpretować jako częstotliwości graniczne (pogląd częstotliwościowy), ale zastosowania nie ograniczają się do tego przypadku. Warunkowanie prawdopodobieństw i reguła Bayesa to główne narzędzia w obliczaniu prawdopodobieństw a posteriori z prawdopodobieństw poprzednich. Na przykład, biorąc pod uwagę początkową sekwencję binarną $x_1...x_{n-1}$, jakie jest prawdopodobieństwo, że następny bit będzie 1? Prawdopodobieństwo zaobserwowania x_n w czasie n, biorąc pod uwagę wcześniejsze obserwacje $x_1...x_{n-1}$, można obliczyć za pomocą reguły mnożenia lub reguły łańcuchowej, jeśli znany jest prawdziwy rozkład generujący μ ciągów $x_1x_2x_3...$:

$\mu(x_{<n}\underline{x}_n)=\mu(\underline{x}_{1:n})/\mu(\underline{x}_{<n})$). Problem polega jednak na tym, że często nie znamy prawdziwego rozkładu μ (np. w przypadku prognozowania pogody i giełdy). Subiektywista używa prawdopodobieństw do scharakteryzowania stopnia wiary agenta w coś (lub jego wiarygodności), a nie do scharakteryzowania fizycznych procesów losowych. Jest to najbardziej odpowiednia interpretacja prawdopodobieństw w AI. Jest nieco zaskakujące, że można wykazać, że prawdopodobieństwa również respektują aksjomaty prawdopodobieństwa Kolmogorowa i regułę łańcuchową dla prawdopodobieństw warunkowych, zakładając tylko kilka prawdopodobnych reguł jakościowych, których powinny przestrzegać. Stąd, jeśli prawdopodobieństwo $x_{1:n}$ wynosi $\xi(\underline{x}_{1:n})$, stopień wiary w x_n przy założeniu $x_{<n}$ jest, ponownie, dany przez prawdopodobieństwo warunkowe: $\xi(x_{<n}\underline{x}_n)=\xi(\underline{x}_{1:n})/\xi(\underline{x}_{<n})$. Reguła łańcuchowa pozwala na określenie prawdopodobieństw a posteriori/prawdopodobieństw z prawdopodobieństw wcześniejszych, ale pozostawia otwartą kwestię, jak określić same prawdopodobieństwa wcześniejsze. W fizyce statystycznej zasada obojętności (zasada symetrii) i zasada maksymalnej entropii mogą być często wykorzystywane do określania prawdopodobieństw wcześniejszych, ale tylko brzytwa Ockhama jest wystarczająco ogólna, aby przypisać prawdopodobieństwa wcześniejsze w każdej sytuacji, zwłaszcza w celu poradzenia sobie ze złożonymi domenami typowymi dla sztucznej inteligencji.

Algorytmiczne prawdopodobieństwo i uniwersalna indukcja

Brzytwa Ockhama (odpowiednio zinterpretowana i będąca w kompromisie z zasadą obojętności Epikura) mówi nam, aby przypisywać wysoką/niską prawdopodobieństwo a priori prostym/złożonym ciągom x. Używając K jako miary złożoności, każda monotoniczna funkcja malejąca K, np. $\xi(\underline{x})=2^{-K(x)}$, spełniałaby to kryterium. Ale ξ musi również spełniać aksjomaty prawdopodobieństwa, więc musimy być nieco ostrożniejsi. Solomonoff zdefiniował uniwersalne wcześniejsze $\xi(\underline{x})$ jako prawdopodobieństwo, że wyjście uniwersalnej maszyny Turinga U zaczyna się od x, gdy na taśmie wejściowej podano uczciwe rzuty monetą. Formalnie ξ można zdefiniować jako

$$\xi(\underline{x}) \;:=\; \sum_{p\,:\,U(p)=x*} 2^{-l(p)} \;\geq\; 2^{-K(x)}, \qquad (16)$$

gdzie suma jest po wszystkich (tzw. minimalnych) programach p, dla których U wyprowadza ciąg zaczynający się od x. Nierówność wynika z pominięcia wszystkich wyrazów w p z wyjątkiem najkrótszego p obliczającego x. Ściśle rzecz biorąc, ξ jest tylko półmiarą, ponieważ nie jest znormalizowane do 1, ale jest to dopuszczalne/korygowalne. Wyprowadzamy następującą granicę:

$$\sum_{t=1}^{\infty}(1-\xi(x_{<t}\underline{x}_t))^2 \;\leq\; -\tfrac{1}{2}\sum_{t=1}^{\infty}\ln\xi(x_{<t}\underline{x}_t) \;=\; -\tfrac{1}{2}\ln\xi(\underline{x}_{1:\infty}) \;\leq\; \tfrac{1}{2}\ln 2\cdot K(x_{1:\infty})$$

W pierwszej nierówności użyliśmy $(1-a)^2\leq -1/2\ \ln a$ dla $0\leq a\leq 1$. W równości zamieniliśmy sumę na logarytm i wyeliminowaliśmy otrzymany iloczyn za pomocą reguły łańcuchowej (6). W ostatniej nierówności użyliśmy (16). Jeśli $x_{1:\infty}$ jest ciągiem obliczalnym, to $K(x_{1:\infty})$ jest skończony, co implikuje $\xi(x_{<t}\underline{x}_t)\to 1(\sum_{t=1}^{\infty}(1-a_t)^2<\infty\Rightarrow a_t\to 1)$. Oznacza to, że jeśli otoczenie jest ciągiem obliczalnym (którymkolwiek, np. cyframi π lub e w reprezentacji binarnej), po zobaczeniu pierwszych kilku cyfr ξ prawidłowo przewiduje następną cyfrę z dużym prawdopodobieństwem, tj. rozpoznaje strukturę ciągu. Załóżmy teraz, że prawdziwa sekwencja jest wylosowana z rozkładu μ, tj. prawdziwe (obiektywne) prawdopodobieństwo $x_{1:n}$ wynosi $\mu(\underline{x}_{1:n})$, ale μ jest nieznane. W jaki sposób późniejsze

(subiektywne) przekonanie $\xi(x_{<n}\underline{x}_n) = \xi(\underline{x}_n)/\xi(\underline{x}_{<n})$ jest powiązane z prawdziwym (obiektywnym) prawdopodobieństwem późniejszym μ(x<nxn) $\mu(x_{<n}\underline{x}_n)$. Kluczowym wynikiem Solomonoffa jest to, że późniejsze (subiektywne) przekonania zbiegają się do prawdziwych (obiektywnych) prawdopodobieństw późniejszych, jeśli te ostatnie są obliczalne. Dokładniej, wykazał, że

$$\sum_{t=1}^{\infty}\sum_{x_{<t}}\mu(\underline{x}_{<t})\Big(\xi(x_{<t}\underline{0}) - \mu(x_{<t}\underline{0})\Big)^2 \stackrel{+}{\leq} \tfrac{1}{2}\ln 2 \cdot K(\mu). \qquad (17)$$

K(μ) jest skończone, jeśli μ jest obliczalne, ale suma nieskończona na lewej półsłupku może być skończona tylko wtedy, gdy różnica $\xi(x_{<t}\underline{0}) - \mu(x_{<t}\underline{0})$ dąży do zera dla t→∞ z μ-prawdopodobieństwem 1. Pokazuje to, że użycie ξ jako oszacowania μ może być rozsądnym rozwiązaniem.

Granice strat i optymalność Pareto

Większość przewidywań jest ostatecznie wykorzystywana jako podstawa do podjęcia decyzji lub działania, które samo w sobie prowadzi do pewnej nagrody lub straty. Niech $l_{xtyt} \in [0,1] \subset IR$ będzie otrzymaną stratą podczas wykonywania przewidywania/decyzji/działania $y_t \in Y$, a $x_t \in X$ będzie t-tym symbolem sekwencji. Niech $y^{\Lambda}_t \in Y$ będzie przewidywaniem (przyczynowego) schematu przewidywania Λ. Prawdziwe prawdopodobieństwo, że następnym symbolem będzie x_t, przy założeniu $x_{<t}$, wynosi $\mu(x_{<t}\underline{x}_t)$. Oczekiwana strata przy przewidywaniu y_t wynosi $E[l_{xtyt}]$. Całkowita μ-oczekiwana strata poniesiona przez schemat Λ w pierwszych n przewidywaniach wynosi

$$L_n^{\Lambda} := \sum_{t=1}^{n} \mathbf{E}[\ell_{x_t y_t^{\Lambda}}] = \sum_{t=1}^{n}\sum_{x_{1:t}\in\mathcal{X}^t} \mu(\underline{x}_{1:t})\ell_{x_t y_t^{\Lambda}}. \qquad (18)$$

Na przykład dla straty błędu $l_{xy} = 1$ jeśli x=y i 0 w przeciwnym wypadku, L^{Λ}_n jest oczekiwaną liczbą błędów predykcji, którą oznaczamy jako E^{Λ}_n. Celem jest zminimalizowanie oczekiwanej straty. Bardziej ogólnie, definiujemy schemat predykcji sekwencji Λ_ρ (później nazywany również SP_ρ) $y_t^{\Lambda\rho} := \operatorname{argmin}_{y_t\in\mathcal{Y}}\sum_{x_t}\rho(x_{<t}\underline{x}_t)\ell_{x_t y_t}$, który minimalizuje stratę oczekiwaną ρ. Jeśli μ jest znane, Λ_μ jest oczywiście najlepszym schematem predykcji w sensie osiągnięcia minimalnej straty oczekiwanej ($L^{\Lambda\mu}_n \leq L^{\Lambda}_n$ dla dowolnego Λ). Można udowodnić następujące ograniczenie straty dla uniwersalnego predyktora Λ_ξ

$$0 \leq L_n^{\Lambda\xi} - L_n^{\Lambda\mu} \leq 2\ln 2 \cdot K(\mu) + 2\sqrt{L_n^{\Lambda\mu}\ln 2 \cdot K(\mu)}. \qquad (19)$$

Razem z $L_n \leq n$ pokazuje to, że $1/nL^{\Lambda\xi}_n - 1/nL^{\Lambda\mu}_n = O(n^{-1/2})$, tj. asymptotycznie Λ_ξ osiąga optymalną średnią stratę Λ_μ z szybką zbieżnością. Ponadto $L^{\Lambda\xi}{}_\infty$ jest skończone, jeśli $L^{\Lambda\mu}{}_\infty$ jest skończone i $L^{\Lambda\xi}_n / L^{\Lambda\mu}_n \to 1$, jeśli $L^{\Lambda\mu}{}_\infty$ nie jest skończone. Ograniczenie (19) implikuje również $L_n^{\Lambda} \geq L_n^{\Lambda\xi} - 2\sqrt{L_n^{\Lambda\xi}\ln 2 \cdot K(\mu)}$, co pokazuje, że żaden (przyczynowy) predyktor Λ nie osiąga znacząco mniejszej (oczekiwanej) straty niż Λ_ξ. W świetle tych wyników można uczciwie stwierdzić, że pomijając kwestie obliczeniowe, problem przewidywania sekwencji został rozwiązany w sposób uniwersalny. Innym rodzajem optymalności jest optymalność Pareto. Uniwersalna wcześniejsza ξ jest optymalna w sensie Pareto w tym sensie, że nie ma innego predyktora, który prowadzi do równej lub mniejszej straty we wszystkich środowiskach. Każda poprawa osiągnięta przez jakiś predyktor Λ w stosunku do Λ_ξ w niektórych środowiskach jest równoważona przez pogorszenie w innych środowiskach.

Uniwersalny agent algorytmiczny AIXI

Aktywnych systemów, takich jak granie w gry (SG) i optymalizacja (FM), nie można sprowadzić do systemów indukcyjnych. Głównym pomysłem tej pracy jest uogólnienie uniwersalnej indukcji do ogólnego modelu agenta opisanego w sekcji 2. W tym celu uogólniamy ξ, aby uwzględnić działania jako warunki i zastępujemy μ przez ξ w racjonalnym modelu agenta, co skutkuje modelem AIξ(=AIXI). W ten sposób rozwiązuje się problem, że prawdziwe prawdopodobieństwo a priori μ jest zwykle nieznane. Można pokazać zbieżność ξ→μ, co wskazuje, że model AIξ mógłby zachowywać się optymalnie w dowolnym obliczalnym, ale nieznanym środowisku ze sprzężeniem zwrotnym wzmocnienia. Głównym celem tej sekcji jest zbadanie, czego możemy oczekiwać od uniwersalnie optymalnego agenta i wyjaśnienie znaczenia uniwersalny, optymalny itd. Niestety ograniczenia podobne do ograniczenia strat (19) w przypadku SP nie mogą obowiązywać dla żadnego aktywnego agenta. Zmusza nas to do obniżenia naszych oczekiwań co do uniwersalnych optymalnych agentów i wprowadzenia innych (słabszych) miar wydajności. Na koniec pokazujemy, że AIξ jest optymalny w sensie Pareto w tym sensie, że nie ma innej polityki dającej wyższą lub równą wartość we wszystkich środowiskach i ściśle wyższą wartość w co najmniej jednym.

Uniwersalny model AIξ

Definicja modelu AIξ. Opracowaliśmy wystarczająco dużo formalizmu, aby zaproponować nasz uniwersalny model AIξ. Wszystko, co musimy zrobić, to odpowiednio uogólnić uniwersalną półmiarę ξ z poprzedniej sekcji i zastąpić prawdziwe, ale nieznane prawdopodobieństwo a priori μ^{AI} w modelu AIμ tym uogólnionym ξ^{AI}. W jakim sensie ten model AIξ jest uniwersalny, zostanie omówione później. W formulacji funkcjonalnej definiujemy uniwersalne prawdopodobieństwo ξ^{AI} środowiska q tak samo jak $2^{-l(q)}$:

$$\xi(q) \;:=\; 2^{-l(q)}$$

Definicja nie mogłaby być prostsza. Zbierając wzory z sekcji 2.4 i zastępując μ(q) przez ξ(q) otrzymujemy definicję agenta AIξ w formie funkcjonalnej. Biorąc pod uwagę historię $\dot{y}\dot{x}_{<k}$, polityka p^{ξ} funkcjonalnego agenta AIξ jest dana przez

$$\dot{y}_k \;:=\; \arg\max_{y_k} \max_{p:p(\dot{x}_{<k})=\dot{y}_{<k}y_k} \sum_{q:q(\dot{y}_{<k})=\dot{x}_{<k}} 2^{-l(q)} \cdot V^{pq}_{km_k} \qquad (20)$$

w cyklu k, gdzie V^{pq}_{kmk} jest całkowitą nagrodą cykli k do m_k, gdy agent p wchodzi w interakcję ze środowiskiem q. Usunęliśmy mianownik $\Sigma_q\mu(q)$ z (2), ponieważ jest on niezależny od $p\in\dot{P}_k$, a stały mnożnik nie zmienia się $\arg\max_{yk}$. W przypadku iteracyjnej formulacji uniwersalne prawdopodobieństwo ξ można uzyskać, wstawiając funkcjonał ξ(q) do (13):

$$\xi(y\underline{x}_{1:k}) \;=\; \sum_{q:q(y_{1:k})=x_{1:k}} 2^{-l(q)}. \qquad (21)$$

Zastępując μ przez ξ w (11) iteracyjny agent AIξ generuje

$$\dot{y}_k \equiv \dot{y}^{\xi}_k := \arg\max_{y_k}\sum_{x_k}\max_{y_{k+1}}\sum_{x_{k+1}}\ldots\max_{y_{m_k}}\sum_{x_{m_k}}(r(x_k)+\ldots+r(x_{m_k}))\cdot\xi(\dot{y}\dot{x}_{<k}y\underline{x}_{k:m_k}) \qquad (22)$$

w cyklu k przy danej historii $\dot{y}\dot{x}_{<k}$. Równoważność funkcjonalnego i iteracyjnego modelu AI (twierdzenie 2) jest prawdziwa dla każdej chronologicznej półmiary ρ, szczególnie dla ξ, stąd możemy mówić o modelu AIξ pod tym względem. Zależy to (w niewielkim stopniu) od wyboru uniwersalnej maszyny Turinga. $l(\langle q \rangle)$ jest zdefiniowane tylko do stałej addytywnej. Model AIξ zależy również od wyboru X =R×O i Y, ale nie spodziewamy się żadnego odchylenia, gdy przestrzenie są wybrane wystarczająco proste, np. wszystkie ciągi o długości 2^{16}. Wybranie IN jako przestrzeni słów byłoby idealne, ale czy w tym przypadku istnieją maksima (suprema), należy wcześniej wykazać. Jedyną nietrywialną zależnością jest zależność od funkcji horyzontu m_k, która zostanie omówiona później. Tak więc poza m_k i nieistotnymi szczegółami agent AIξ jest jednoznacznie zdefiniowany. Nie zależy ona od żadnych założeń dotyczących środowiska, poza tym, że jest generowana przez pewien obliczalny (ale nieznany!) rozkład prawdopodobieństwa.

Konwergencja ξ do μ. Podobnie jak w przypadku (17) można pokazać, że μ-oczekiwana różnica kwadratowa μ i ξ jest skończona dla obliczalnego μ. To z kolei pokazuje, że $\xi(yx_{<k}y\underline{x}_k)$ szybko zbiega do $\mu(yx_{<k}y\underline{x}_k)$ dla k→∞z μ-prawdopodobieństwem 1. Tok rozumowania jest taki sam; y są czystymi widzami. Zmieni się to, gdy przeanalizujemy granice strat/nagród analogicznie do (19). Bardziej ogólnie można pokazać [30], że

$$\xi(yx_{<k}y\underline{x}_{k:m_k}) \stackrel{k\to\infty}{\longrightarrow} \mu(yx_{<k}y\underline{x}_{k:m_k}). \qquad (23)$$

Daje to nadzieję, że wyniki $\dot{y}_k$ modelu AIξ (22) mogą zbiegać się do wyników $\dot{y}_k$ modelu AIμ (11). Chcemy nazwać model AI uniwersalnym, jeśli jest μ-niezależny (nieobciążony, wolny od modelu) i jest w stanie rozwiązać każdy rozwiązywalny problem i nauczyć się każdego zadania, którego można się nauczyć. Ponadto, nazywamy model uniwersalny uniwersalnie optymalnym, jeśli nie ma programu, który mógłby rozwiązać lub nauczyć się znacznie szybciej (w kategoriach cykli interakcji). Rzeczywiście, model AIξ jest wolny od parametrów, ξ zbiega się do μ (23), model AIμ jest sam w sobie optymalny i nie spodziewamy się, że żaden inny model nie zbiegnie się szybciej do AIμ przez analogię do SP (19):

Twierdzenie (Spodziewamy się, że AIXI będzie uniwersalnie optymalny). To jest nasze główne twierdzenie. W pewnym sensie intencją pozostałych sekcji jest bardziej rygorystyczne zdefiniowanie tego stwierdzenia i udzielenie dalszego wsparcia. Relacja porządku inteligencji. Definiujemy ξ-oczekiwaną nagrodę w cyklach k do m polityki p podobnie jak (2) i (20). Rozszerzamy definicję na programy $p \notin \dot{P}_k$, które nie są zgodne z bieżącą historią.

$$V_{km}^{p\xi}(\dot{y}\dot{x}_{<k}) := \frac{1}{\mathcal{N}} \sum_{q:q(\dot{y}_{<k})=\dot{x}_{<k}} 2^{-l(q)} \cdot V_{km}^{\tilde{p}q}. \qquad (24)$$

Normalizacja N jest znowu konieczna tylko do interpretacji V_{km} jako oczekiwanej nagrody, ale poza tym jest niepotrzebna. Dla spójnych polityk $p \in \dot{P}_k$ definiujemy $\tilde{p}:=p$. Dla $p \notin \dot{P}_k$: $\tilde{p}$ jest modyfikacją p w taki sposób, że jej wyniki są zgodne z bieżącą historią $\dot{y}\dot{x}_{<k}$, stąd $\tilde{p} \in \dot{P}_k$, ale niezmienione dla bieżących i przyszłych cykli ≥k. Używając tej definicji V_{km} moglibyśmy wziąć maksimum dla wszystkich polityk p w (20), a nie tylko dla tych spójnych.

Definicja 5 (Relacje porządku inteligencji). Politykę p nazywamy bardziej lub równie inteligentną niż p' i zapisujemy

$$p \succeq p' \quad :\Leftrightarrow \quad \forall k \forall \dot{y}\dot{x}_{<k} : V_{km_k}^{p\xi}(\dot{y}\dot{x}_{<k}) \geq V_{km_k}^{p'\xi}(\dot{y}\dot{x}_{<k}).$$

tj. jeśli p w jakichkolwiek okolicznościach daje wyższą ξ-oczekiwaną nagrodę niż p'.

Ponieważ algorytm $p*$ za agentem AIξ maksymalizuje $V^{p\xi}_{kmk}$ mamy pξ ⩾ p dla wszystkich p. Model AIξ jest zatem najinteligentniejszym agentem w.r.t. ⩾. Relacja ⩾ jest uniwersalną relacją porządku w tym sensie, że jest wolna od jakichkolwiek parametrów (oprócz m_k) lub określonych założeń dotyczących środowiska. Dowód, ⩾ czyli wiarygodny porządek inteligencji (który uważamy za prawdziwy), udowodniłby, że AIξ jest uniwersalnie optymalny. Możemy dalej zapytać: Jak przydatny jest ⩾ do porządkowania polityk o praktycznym znaczeniu z inteligencją pośrednią lub jak może pomóc ⩾ w kierowaniu w kierunku konstruowania bardziej inteligentnych systemów z rozsądnym czasem obliczeniowym?

O optymalności AIXI

W tej sekcji przedstawiamy sposoby na dowód optymalności AIXI. Źródłami inspiracji są granice strat SP udowodnione w sekcji 3 oraz kryteria optymalności z literatury na temat sterowania adaptacyjnego (głównie) dla układów liniowych . Oczekuje się, że granice wartości dla AIXI będą, w pewnym sensie, słabsze niż granice strat SP, ponieważ klasa problemów objętych przez AIXI jest znacznie większa niż klasa problemów indukcyjnych. Zbieżność ξ do μ została już udowodniona, ale nie jest wystarczająca, aby ustalić zbieżność zachowania modelu AIXI z zachowaniem modelu AIμ. Skupimy się na trzech podejściach do ogólnego dowodu optymalności: Znaczenie „uniwersalnej optymalności". Pierwszym krokiem jest zbadanie, czego możemy oczekiwać od AIXI, tj. co oznacza uniwersalna optymalność. „Uczący się" (taki jak AIXI) może zbiegać się do optymalnego świadomego decydenta (takiego jak AIμ) w kilku znaczeniach. Możliwe istotne koncepcje ze statystyki to: spójność, samostrojenie, samooptymalizacja, wydajność, bezstronność, asymptotyczna lub skończona zbieżność , optymalność Pareto i niektóre inne. Niektóre koncepcje są silniejsze niż to konieczne, inne słabsze niż pożądane, ale odpowiednie na początek. Samooptymalizacja jest definiowana jako asymptotyczna zbieżność średniej prawdziwej wartości $\frac{1}{m}V_{1m}^{p^\xi\mu}$ do optymalnej wartości $\frac{1}{m}V_{1m}^{*\mu}$. Oprócz szybkości zbieżności, samooptymalizacja AIXI najbardziej odpowiadałaby granicom strat udowodnionym dla SP. Badamy, które własności są pożądane i w jakich okolicznościach model AIXI spełnia te właściwości. Pokażemy, że żaden uniwersalny model, w tym AIXI, nie może być ogólnie samooptymalizujący. Odwrotnie, pokazujemy, że AIXI jest optymalny w sensie Pareto w tym sensie, że nie ma innej polityki, która działałaby lepiej lub równie dobrze we wszystkich środowiskach, a ściślej lepiej w co najmniej jednym.

Ograniczone klasy środowiskowe. Problem definiowania i udowadniania ogólnych granic wartości staje się bardziej wykonalny, gdy w pierwszym kroku weźmiemy pod uwagę ograniczone klasy pojęć. Analizujemy AIXI dla znanych klas (takich jak środowiska markowskie lub faktoryzowalne), a zwłaszcza dla nowych klas (zapominajkowych, istotnych, asymptotycznie uczących się, dalekowzrocznych, jednolitych, pseudopasywnych i pasywnych) zdefiniowanych później .

Uogólnienie AIXI na ogólne mieszaniny Bayesa. Innym podejściem jest uogólnienie AIXI na AIζ, gdzie $\zeta() = \Sigma_{\nu \in M} w_\nu \nu()$ jest ogólną mieszaniną Bayesa rozkładów ν w pewnej klasie M. Jeśli M jest zbiorem wieloprzeliczalnych półmiar wyliczonych przez maszynę Turinga, wówczas AIζ pokrywa się z AIXI. Jeśli M jest (wielorakim) zbiorem pasywnych efektywnych środowisk, wówczas AIXI redukuje się do predyktora Λ_ξ, który, jak wykazano, działa dobrze. Można wykazać, że te granice strat/wartości uogólniają się na szersze klasy, przynajmniej asymptotycznie. W szczególności dla ergodycznych mdps pokazaliśmy, że AIζ jest samooptymalizujące. Oczywiście, najmniej, czego musimy wymagać od M, aby mieć szansę na znalezienie samooptymalizującej się polityki, to aby w ogóle istniała jakaś

samooptymalizująca się polityka. Kluczowym wynikiem jest to, że ten konieczny warunek jest również wystarczający. Bardziej ogólnie, kluczem nie jest udowodnienie wyników bezwzględnych dla określonych klas problemów, ale udowodnienie wyników względnych w postaci „jeśli istnieje polityka o pewnych pożądanych właściwościach, to AIζ również posiada te pożądane właściwości". Jeśli istnieją zadania, których nie można rozwiązać za pomocą żadnej polityki, AIζ nie można obwiniać za niepowodzenie. Klasy środowiskowe, które umożliwiają samooptymalizujące się polityki, obejmują bandytów, procesy i.i.d., zadania klasyfikacyjne, pewne klasy pompp, ergodyczne mdp k-tego rzędu, środowiska faktoryzowalne, powtarzalne gry i problemy predykcyjne. Należy zauważyć, że w tym podejściu dla każdej klasy środowiskowej mamy odpowiadający jej model AIζ, podczas gdy w podejściu realizowanym w tym artykule ten sam uniwersalny model AIXI jest analizowany dla wszystkich klas środowiskowych. Optymalność przez konstrukcję. Możliwym dalszym podejściem do „dowodu" optymalności jest uznanie AIXI za optymalne z konstrukcji. Ta perspektywa jest powszechna w różnych (prostszych) ustawieniach. Na przykład w problemach bandytów, gdzie pociągnięcie ramienia i prowadzi do nagrody 1 (0) z nieznanym prawdopodobieństwem p_i $(1-p_i)$, tradycyjnym rozwiązaniem bayesowskim niepewności dotyczącej p_i jest założenie jednolitego (lub beta) prior względem pi i zmaksymalizowanie subiektywnie oczekiwanej sumy nagród w wielu próbach. Dokładne rozwiązanie (w kategoriach indeksów Gittinsa) jest powszechnie uważane za „optymalne", chociaż istnieją uzasadnione alternatywne podejścia. Podobnie, ale prostsze, zakładając jednolity subiektywny prior względem parametru Bernoulliego p(i) ∈[0,1], dochodzi się do rozsądnej, ale bardziej kontrowersyjnej reguły Laplace'a do przewidywania sekwencji i.i.d. AIXI jest podobne w tym sensie, że nieznane μ∈M jest analogiem nieznanego p∈[0,1], a wcześniejsze przekonania $w_\nu = 2^{-K(\nu)}$ uzasadnione brzytwą Ockhama są analogiem rozkładu jednostajnego na [0,1]. W tym samym sensie, co rozwiązanie Gittinsa problemu bandyty i reguła Laplace'a dla ciągów Bernoulliego, AIXI można również uznać za optymalne pod względem konstrukcyjnym. Twierdzenia odnoszące AIXI do AIμ nie byłyby uważane za dowody optymalności AIXI, ale tak samo jak to, o ile trudniej jest operować, gdy μ jest nieznane, tj. osiągnięcia pierwszych trzech podejść są po prostu reinterpretowane.

Granice wartości i koncepcje rozdzielności

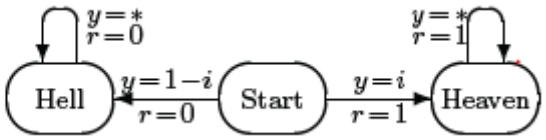
Wprowadzenie. Wartości V_{km} związane z systemami AI odpowiadają mniej więcej ujemnej stracie $-L^{\Lambda}_n$ systemów SP. W SP interesowały nas małe ograniczenia dla nadmiaru strat $L^{\Lambda\xi}_n - L^{\Lambda}_n$. Niestety, proste ograniczenia wartości dla AIξ w kategoriach V_{km} analogiczne do ograniczenia strat (19) nie obowiązują. Mamy nawet trudności ze sprecyzowaniem, czego możemy się spodziewać dla AIξ lub dowolnego systemu AI, który twierdzi, że jest uniwersalnie optymalny. W konsekwencji nie możemy mieć dowodu, jeśli nie wiemy, co udowodnić. W SP jedyną ważną własnością μ dla udowodnienia ograniczeń strat była jego złożoność K(μ). Zobaczymy, że w przypadku AI nie ma użytecznych ograniczeń w kategoriach K(μ). Musimy albo badać ograniczone klasy problemów, albo rozważać ograniczenia zależne od innych własności μ, a nie tylko od jego złożoności. W dalszej części przedstawimy trudności na dwóch przykładach i wprowadzimy koncepcje, które mogą być przydatne do udowodnienia ograniczeń wartości. Pomimo trudności w nawet roszczeniu sobie użytecznych granic wartości, mimo wszystko, jesteśmy przekonani, że relacja porządku (Definicja 5) poprawnie formalizuje intuicyjne znaczenie inteligencji, a zatem, że agent AIξ jest uniwersalnie optymalny

(Pseudo) Pasywny μ i przykład HeavenHell. W poniższym przykładzie wybieramy $m_k = m$. Chcemy porównać prawdziwą, tj. μ-oczekiwaną wartość V^{μ}_{1m} μ-niezależnej uniwersalnej polityki p^{best} z dowolną inną polityką p. Naiwnie, moglibyśmy oczekiwać istnienia polityki p^{best}, która maksymalizuje V^{μ}_{1m}, poza poprawkami addytywnymi niższego rzędu dla $m \to \infty$:

$$V_{1m}^{p^{best}\mu} \geq V_{1m}^{p\mu} - o(...) \quad \forall \mu, p \qquad (25)$$

Takie polityki są czasami nazywane samooptymalizującymi . Należy zauważyć, że $V^{p\mu\mu}_{1m} \geq V^{p\mu}_{1m}$ ∀p, ale p^{μ} nie jest kandydatem na (uniwersalne) p^{best}, ponieważ zależy od μ. Z drugiej strony polityka p^{ξ} agenta AIξ maksymalizuje V^{ξ}_{1m} z definicji $(p^{\xi} \succeq p)$. Ponieważ V^{ξ}_{1m} jest uważane za przypuszczenie V^{μ}_{1m}, możemy oczekiwać, że $p^{best} = p^{\xi}$ w przybliżeniu zmaksymalizuje V^{μ}_{1m}, tj. (25) będzie spełnione. Rozważmy klasę problemu (zbiór środowisk) $M=\{\mu_0,\mu_1\}$ z Y =R={0,1} i $r_k = \delta_{iy1}$ w środowisku μ_i, gdzie symbol Kroneckera δ_{xy} jest zdefiniowany jako 1 dla x=y i 0 w przeciwnym razie. Pierwsza akcja y_1 decyduje, czy pójdziesz do nieba ze wszystkimi przyszłymi nagrodami r_k wynoszącymi 1 (dobrze) czy do piekła ze wszystkimi przyszłymi nagrodami wynoszącymi 0 (źle). Zauważ, że μ_i to (deterministyczne, nieergodyczne) mdps:

μ_i =



Jest oczywiste, że jeśli μ_i, tj. i jest znane, optymalna polityka $p^{\mu i}$ to wyprowadzenie y_1=i w pierwszym cyklu z $V^{p\mu i\mu}_{1m}$ = m. Z drugiej strony, każda nieobciążona polityka p^{best} niezależna od rzeczywistego μ wyprowadza albo y_1=1, albo y_1=0. Niezależnie od rzeczywistego wyboru y_1, zawsze istnieje środowisko ($\mu=\mu_{1-y1}$), dla którego ten wybór jest katastrofalny ($V^{pbest\mu}_{1m}$ = 0). Żaden pojedynczy agent nie może działać dobrze w obu środowiskach μ_0 i μ_1. Prawa strona równania (25) jest równa m−o(m) dla $p=p^{\mu}$. Dla każdego p^{best} istnieje μ, dla którego lewa strona równania jest równa zero. Pokazaliśmy, że żadne p^{best} nie może spełnić (25) dla wszystkich μ i p, więc nie możemy oczekiwać, że p^{ξ} to zrobi. Niemniej jednak istnieją klasy problemów, dla których zachodzi (25), na przykład SP. Dla SP, (25) jest po prostu reformulacją (19) z odpowiednim wyborem dla p^{best}, mianowicie Λ_{ξ} (które różni się od p^{ξ}). Oczekujemy, że (25) będzie spełnione dla wszystkich problemów indukcyjnych, w których środowisko nie jest pod wpływem wyjścia agenta. Chcemy nazwać te μ, środowiskami pasywnymi lub indukcyjnymi. Ponadto chcemy nazwać M i μ∈M spełniającymi (25) z $p^{best} = p^{\xi}$ pseudopasywnymi. Oczekujemy więc, że indukcyjne μ będzie pseudopasywne.

Przykład OnlyOne. Podajmy kolejny przykład, aby zademonstrować trudności w ustalaniu granic wartości. Niech X =R={0,1} i |Y| będą duże. Rozważamy wszystkie (deterministyczne) środowiska, w których pojedyncze złożone wyjście y^* jest poprawne (r=1), a wszystkie inne są błędne (r=0). Klasa problemów M jest zdefiniowana przez

$$\mathcal{M} := \{\mu_{y^\star} : y^* \in \mathcal{Y},\ K(y^*) =_{\lfloor} \log|\mathcal{Y}|_{\rfloor}\}, \quad \text{where} \quad \mu_{y^\star}(yx_{<k}y_k\underline{1}) := \delta_{y_k y^\star}\ \forall k.$$

Istnieje $N \stackrel{\times}{=} |\mathcal{Y}|$ takich y^*. Jedynym sposobem, w jaki μ-niezależna polityka p może znaleźć poprawne y^*, jest wypróbowanie jednego y po drugim w określonej kolejności. W pierwszych N−1 cyklach testowanych jest co najwyżej N−1 różnych y. Ponieważ istnieje N różnych możliwych $y*$, zawsze istnieje μ∈M, dla którego p daje błędne wyniki w pierwszych N−1 cyklach. Liczba błędów wynosi $E^p_\infty \geq N-1 \stackrel{\times}{=} |\mathcal{Y}| \stackrel{\times}{=} 2^{K(y^\star)} \stackrel{\times}{=} 2^{K(\mu)}$ dla tego μ. Ponieważ jest to prawdą dla dowolnego p, jest to również prawdą dla modelu AIξ, stąd $E^{p^{\xi}}_k \leq 2^{K(\mu)}$ jest najlepszą możliwą granicą błędu, jakiej możemy się spodziewać, zależną tylko od K(μ). Właściwie wyprowadzimy taką granicę w rozdz. 5.1 dla środowisk indukcyjnych. Niestety, ponieważ interesuje nas głównie obszar cyklu $k \ll |\mathcal{Y}| \stackrel{\times}{=} 2^{K(\mu)}$, to

ograniczenie jest puste. Nie ma interesujących ograniczeń dla deterministycznego μ zależnych tylko od K(μ), w przeciwieństwie do przypadku SP. Granice muszą albo zależeć od dodatkowych własności μ, albo musimy rozważyć wyspecjalizowane ograniczenia dla ograniczonych klas problemów. Przypadek probabilistycznego μ jest podobny. Podczas gdy dla SP istnieją użyteczne ograniczenia w kategoriach $L^{\wedge\mu}_k$ i K(μ), nie ma takich ograniczeń dla AIξ. Ponownie, nie jest to wada AIξ, ponieważ dla żadnego nieobciążonego systemu AI błędy/nagrody nie mogłyby być ograniczone w kategoriach K(μ), a błędy/nagrody tylko AIμ. Istnieje sposób na wykorzystanie ograniczeń brutto (np. $2^{K(\mu)}$). Załóżmy, że po rozsądnej liczbie cykli k, informacja $\dot{x}_{<k}$ postrzegana przez agenta AIξ zawiera wiele informacji o prawdziwym środowisku μ. Informacja w $\dot{x}_{<k}$ może być zakodowana w dowolnej formie. Załóżmy, że złożoność $K(\mu|\dot{x}_{<k})$ pod warunkiem, że $\dot{x}_{<k}$ jest znane, jest rzędu 1. Rozważmy twierdzenie, ograniczające sumę nagród lub innych wielkości w cyklach 1...∞ w kategoriach f(K(μ)) dla funkcji f z f(O(1))=O(1), takiej jak $f(n) = 2^n$. Wówczas będzie istniało ograniczenie dla cykli k...∞ w kategoriach $\approx f(K(\mu|\dot{x}_{<k})) = O(1)$. Stąd ograniczenie takie jak $2^{K(\mu)}$ można zastąpić małym ograniczeniem $\approx 2^{K(\mu|\dot{x}_{<k})} = O(1)$ po k cyklach. Wszystko, co trzeba pokazać/zapewnić/założyć, to że wystarczająca ilość informacji o μ jest przedstawiona (w dowolnej formie) w pierwszych k cyklach. W ten sposób nawet grube ograniczenie może stać się przydatne.Używamy podobnego argumentu, aby udowodnić, że AIξ jest w stanie uczyć się nadzorowanego.

Asymptotyczna uczenie się. W dalszej części osłabiamy (25) w nadziei na uzyskanie ograniczenia stosowalnego do szerszych klas problemów niż pasywne. Rozważ sekwencję wejścia/wyjścia $\dot{y}_1\dot{x}_1...\dot{y}_n\dot{x}_n$ spowodowaną przez AIξ. Na historii $\dot{y}\dot{x}_{<k}$. AIξ wyprowadzi $\dot{y}_k \equiv \dot{y}^{\xi}_k$ w cyklu k. Porównajmy to z $\dot{y}^{\mu}_k$, które wyprowadziłby AIμ, nadal na tej samej historii $\dot{y}\dot{x}_{<k}$ wytworzonej przez AIξ. Ponieważ AIμ maksymalizuje wartość oczekiwaną μ, AIξ powoduje niższe (lub co najwyżej równe) V^{μ}_{kmk}, jeśli $\dot{y}^{\xi}_k$ różni się od $\dot{y}^{\mu}_k$. Niech $D_{n\mu\xi} := \mathbf{E}[\sum_{k=1}^{n} 1 - \delta_{\dot{y}^{\mu}_k \dot{y}^{\xi}_k}]$ będzie oczekiwaną liczbą μ suboptymalnych wyborów AIξ, tj. wyników różnych od AIμ w pierwszych n cyklach. Można ważyć odbiegające przypadki według ich powagi. W szczególności, gdy oczekiwane nagrody μ $V^{p\mu}_{kmk}$ dla $\dot{y}^{\xi}_k$ i $\dot{y}^{\mu}_k$ są równe lub zbliżone do siebie, należy to uwzględnić w definicji $D_{n\mu\xi}$, np. za pomocą współczynnika wagi $[V^{*\mu}_{km}(\dot{y}\dot{x}_{<k}) - V^{p^{\xi}\mu}_{km}(\dot{y}\dot{x}_{<k})]$. Szczegóły te nie mają znaczenia w dalszej dyskusji jakościowej. Ważną różnicą w stosunku do (25) jest to, że tutaj trzymamy się historii wygenerowanej przez AIξ i uznajemy błędną decyzję za co najwyżej jeden błąd. Błędna decyzja w przykładzie HeavenHell w pierwszym cyklu nie jest już uznawana za utratę m nagród, ale za jedną błędną decyzję. W pewnym sensie jest to bardziej sprawiedliwe. Nie należy obwiniać tych, którzy podejmują jedną błędną decyzję, co do której mają zbyt mało dostępnych informacji, aby podjąć poprawną decyzję. Model AIξ zasługiwałby na miano asymptotycznie optymalnego, gdyby prawdopodobieństwo podjęcia złej decyzji zmierzało do zera, tj. jeśli

$$D_{n\mu\xi}/n \to 0 \quad \text{for} \quad n \to \infty, \quad \text{i.e.} \quad D_{n\mu\xi} \;=\; o(n). \qquad (26)$$

Mówimy, że μ można nauczyć się asymptotycznie (przez AIξ), jeśli (26) jest spełnione. Twierdzimy, że AIξ (dla $m_k \to \infty$) może asymptotycznie nauczyć się każdego problemu μ o istotności, tj. AIξ jest asymptotycznie optymalne. Włączyliśmy kwalifikator istotności, ponieważ nie jesteśmy pewni, czy może istnieć dziwne psucie μ (26), ale spodziewamy się, że te μ będą nieistotne z perspektywy AI. W dziedzinie uczenia się istnieje wiele asymptotycznych twierdzeń o zdolności uczenia się, często niezbyt trudnych do udowodnienia. Tak więc dowód (26) może być również wykonalny. Niestety,

asymptotyczne twierdzenia o zdolności uczenia się są często zbyt słabe, aby były przydatne z praktycznego punktu widzenia. Niemniej jednak wskazują one właściwy kierunek.

Jednorodny μ. Ze zbieżności (23) ξ→μ moglibyśmy oczekiwać, że $V^{p\xi}_{kmk} \to V^{p\mu}_{kmk}$ dla wszystkich p, a zatem moglibyśmy również oczekiwać, że $\dot{y}_k^{\xi}$ zdefiniowane w (22) zbiegnie się do $\dot{y}_k^{\mu}$ zdefiniowanego w (11) dla k→∞. Pierwszy problem polega na tym, że jeśli V_{kmk} dla różnych wyborów y_k są prawie równe, to nawet jeśli $V^{p\xi}_{kmk} \approx V^{p\mu}_{kmk}$, $\dot{y}_k^{\xi} \neq \dot{y}_k^{\mu}$ jest możliwe ze względu na nieciągłość argmax_{yk}. Można to naprawić za pomocą ważonego $D_{n\mu\xi}$, jak opisano powyżej. Bardziej poważny jest drugi problem, który wyjaśniamy dla h_k=1 i X =R={0,1}. Aby $\dot{y}_k^{\xi} \equiv \text{argmax}_{y_k} \xi(\dot{y}\dot{r}_{<k} y_k \underline{1})$ zbiegało do $\dot{y}_k^{\mu} \equiv \text{argmax}_{y_k} \mu(\dot{y}\dot{r}_{<k} y_k \underline{1})$), nie wystarczy wiedzieć, że $\xi(\dot{y}\dot{r}_{<k}\dot{y}\underline{\dot{r}}_k) \to \mu(\dot{y}\dot{r}_{<k}\dot{y}\underline{\dot{r}}_k)$, jak udowodniono w (23). Potrzebujemy zbieżności nie tylko dla prawdziwego wyniku $\dot{y}_k$, ale także dla alternatywnych wyników y_k. $\dot{y}_k^{\xi}$ zbiega się do $\dot{y}_k^{\mu}$, jeśli ξ zbiega się jednostajnie do μ, tj. jeśli oprócz (23)

$$\left|\mu(yx_{<k}y'_k\underline{x'_k}) - \xi(yx_{<k}y'_k\underline{x'_k})\right| \;<\; c\cdot\left|\mu(yx_{<k}y\underline{x}_k) - \xi(yx_{<k}y\underline{x}_k)\right| \quad \forall y'_k x'_k \quad (27)$$

zachodzi dla pewnej stałej c (przynajmniej w μ-oczekiwanym sensie). Nazywamy μ spełniającym (27) jednorodny. Dla jednorodnego μ można pokazać (26) z odpowiednio ważonym $D_{n\mu\xi}$ i ograniczonym horyzontem $h_k<h_{max}$. Niestety istnieją istotne μ, które nie są jednorodne.

Inne koncepcje. Poniżej krótko wspomnimy o kilku dalszych koncepcjach. Markowian μ jest definiowany jako zależny tylko od ostatniego cyklu, tj. $\mu(yx_{<k}y\underline{x}_k) = \mu_k(x_{k-1}y\underline{x}_k)$. Mówimy, że μ jest uogólnionym (l-tego rzędu) markowskim, jeśli $\mu(yx_{<k}y\underline{x}_k) = \mu_k(x_{k-l}yx_{k-l+1:k-1}y\underline{x}_k)$ dla ustalonego l. Ta własność ma pewne podobieństwa do faktoryzowalnego μ zdefiniowanego w (14). Jeżeli dalej $\mu_k \equiv \mu_1 \forall k$, μ nazywamy stacjonarnym. Ponadto, μ (ξ) nazywamy zapominalskim, jeżeli $\mu(yx_{<k}y\underline{x}_k)$ $(\xi(yx_{<k}y\underline{x}_k))$ staje się (s) niezależny(e) od $yx_{<l}$ dla ustalonego l i k→∞ z μ-prawdopodobieństwem 1. Ponadto, mówimy, że μ jest dalekowzroczne, jeżeli istnieje $\lim_{m_k\to\infty} \dot{y}_k^{(m_k)}$. Więcej szczegółów zostanie podanych później, gdzie podajemy również przykład dalekowzrocznego μ, dla którego jednak granica m_k→∞ nie ma sensu.

Podsumowanie. Wprowadziliśmy kilka pojęć, które mogą być przydatne do udowodnienia ograniczeń wartości, w tym zapominalskie, istotne, asymptotycznie przyswajalne, dalekowzroczne, jednorodne, (uogólnione)markowskie, faktoryzowalne i (pseudo)pasywne μ. Posortowaliśmy je tutaj, mniej więcej w kolejności malejącej ogólności. Nazywać je będziemy koncepcjami separowalności. Bardziej ogólne (jak istotne, asymptotycznie przyswajalne i dalekowzroczne) μ będziemy nazywać słabo separowalnymi, bardziej restrykcyjne (jak (pseudo) pasywne i faktoryzowalne) μ będziemy nazywać silnie separowalnymi, ale będziemy używać tych kwalifikatorów w bardziej jakościowym, a nie sztywnym sensie. Inne (nieseparowalne) koncepcje to deterministyczne μ i oczywiście klasa wszystkich chronologicznych μ.

Optymalność Pareto AIξ

Ta podsekcja pokazuje optymalność Pareto AIξ analogiczną do SP. Całkowita μ-oczekiwana nagroda $V^{p\xi}_{\mu}$ polityki p^{ξ} modelu AIξ jest centralnym punktem oceny wydajności AIξ. Wiemy, że istnieją polityki (np. p^{μ} AIμ) o wyższej wartości μ ($V^{*}_{\mu} \geq V^{p\xi}_{\mu}$). Ogólnie rzecz biorąc, każda polityka oparta na

oszacowaniu ρ μ, które jest bliższe μ niż ξ, przewyższa p^{ξ} w środowisku μ, po prostu dlatego, że jest bardziej dostosowana do μ. Z drugiej strony, taki system prawdopodobnie działa gorzej niż p^{ξ} w innych środowiskach. Ponieważ nie wiemy z góry, możemy zapytać, czy istnieje polityka p z lepszą lub równą wydajnością niż p^{ξ} we wszystkich środowiskach ν∈M i ściśle lepszą wydajnością dla jednego ν∈M. To wyraźnie uczyniłoby pξ suboptymalnym. Można pokazać, że nie ma takiego p .

Definicja 6 (optymalność Pareto). Polityka $\tilde{p}$ jest nazywana optymalną w sensie Pareto, jeśli nie ma innej polityki p z $V_\nu^p \geq V_\nu^{\tilde{p}}$ dla wszystkich ν∈M i ścisłą nierównością dla co najmniej jednego ν.

Twierdzenie 5 (optymalność Pareto). AIξ alias p^{ξ} jest optymalny w sensie Pareto.

Optymalność Pareto należy uważać za warunek konieczny dla agenta dążącego do bycia optymalnym. Z praktycznego punktu widzenia znaczny wzrost V dla wielu środowisk ν może być pożądany, nawet jeśli powoduje to niewielki spadek V dla kilku innych ν. Niemożność takiej „zrównoważonej" poprawy jest bardziej wymagającym warunkiem dla pξ niż czysta optymalność Pareto. Wykazano, że AIξ jest również zrównoważoną optymalnością Pareto.

Wybór horyzontu

Jedyną znaczącą arbitralnością w modelu AIξ jest wybór funkcji horyzontu $h_k \equiv mk-k+1$. Omawiamy niektóre wybory, które wydają się naturalne i na końcu podajemy wstępne wnioski. Nie będziemy omawiać ad hoc wyborów hk dla konkretnych problemów . Interesują nas uniwersalne wybory m_k.

Stały horyzont. Jeśli wiadomo, że czas życia agenta wynosi m, co w praktyce jest zawsze duże, ale skończone, to wybór m_k = m poprawnie maksymalizuje oczekiwaną przyszłą nagrodę. Czas życia m zwykle nie jest znany z góry, ponieważ w wielu przypadkach czas, w którym jesteśmy skłonni uruchomić agenta, zależy od jakości jego wyników. Z tego powodu często pożądane jest, aby dobre wyniki nie były opóźniane zbyt mocno, jeśli skutkuje to jedynie marginalnym wzrostem nagrody. Można to uwzględnić, tłumiąc przyszłe nagrody. Jeśli na przykład prawdopodobieństwo przetrwania w cyklu wynosi γ<1, odpowiednie jest tłumienie wykładnicze (dyskonto geometryczne) $r_k := r'_k \cdot \gamma^k$, gdzie r'_k są ograniczone, np. $r'_k \in [0,1]$. Wyrażenie (22) jest zbieżne dla $m_k \to \infty$ w tym przypadku. Jednak nie rozwiązuje to problemu, ponieważ wprowadziliśmy nową dowolną skalę czasu $(1-\gamma)^{-1}$. Każde tłumienie wprowadza skalę czasu. Przyjęcie $\gamma \to 1$ jest podatne na te same problemy, co $mk \to \infty$ w przypadku niezdyskontowanym omówionym poniżej.

Dynamiczny horyzont (dyskontowanie uniwersalne i harmoniczne). Największy horyzont z gwarantowaną skończoną i przeliczalną sumą nagród można uzyskać za pomocą dyskonta uniwersalnego $r_k \rightsquigarrow r_k \cdot 2^{-K(k)}$. To zdyskontowanie skutkuje naprawdę dalekowzrocznym agentem z efektywnym horyzontem, który rośnie szybciej niż jakakolwiek obliczalna funkcja. Jest to podobne do dyskonta prawie harmonicznego $r_k \rightsquigarrow r_k \cdot k^{-(1+\varepsilon)}$, ponieważ $2^{-K(k)} \leq 1/k$ dla większości k i $2^{-K(k)} \geq c/(k \log^2 k)$. Bardziej ogólnie, niezmienny względem skali czasu współczynnik tłumienia $r_k = r'_k \cdot k^{-\alpha}$ wprowadza dynamiczną skalę czasu. W cyklu k wkład cyklu $2^{1/\alpha} \cdot k$ jest tłumiony o współczynnik 1/2 . Efektywny horyzont h_k w tym przypadku wynosi ∼k. Wybór $h_k = \beta \cdot k$ z $\beta \sim 2^{1/\alpha}$ jakościowo modeluje to samo zachowanie. Nie wprowadziliśmy arbitralnej skali czasu m, ale ograniczyliśmy dalekowzroczność do pewnej wielokrotności (lub ułamka) długości bieżącej historii. Unikamy w ten sposób wstępnego wyboru globalnej skali czasu m lub 1 /1−γ. Ten wybór ma pewien urok, ponieważ wydaje się, że ludzie w wieku k lat zazwyczaj nie planują swojego życia na dłużej niż, być może, następne k lat ($\beta_{human} \approx 1$). Z praktycznego punktu widzenia ten model może spełniać wszystkie potrzeby, ale z teoretycznego punktu widzenia czujemy się niekomfortowo z takim ograniczeniem horyzontu od samego początku.

Zauważ, że musimy wybrać β = O(1), ponieważ w przeciwnym razie ponownie wprowadzilibyśmy liczbę β, która musi być uzasadniona. Preferujemy uniwersalną zniżkę $\gamma_k = 2^{-K(k)}$, ponieważ pozwala nam ona, jeśli chcemy, „naśladować" wszystkie inne bardziej zachłanne zachowania w oparciu o inne zniżkę γ_k, wybierając $r_k \in [0, c\cdot\gamma_k] \subseteq [0, 2^{-K(k)}]$.

Nieskończony horyzont. Naiwna granica $m_k \to \infty$ w (22) może okazać się dobrze zdefiniowana, a poprzednia dyskusja zbędna. W dalszej części sugerujemy granicę, która jest zawsze dobrze zdefiniowana (dla skończonego Y). Niech $\dot{y}_k^{(m_k)}$ będzie zdefiniowane jak w (22) z zależnością od m_k wyrażoną jawnie. Ponadto niech $\dot{\mathcal{Y}}_k^{(m)} := \{\dot{y}_k^{(m_k)} : m_k \geq m\}$ będzie zbiorem wyników w cyklu k dla wyborów m_k=m,m+1,m+2,.... Ponieważ $\dot{\mathcal{Y}}_k^{(m)} \supseteq \dot{\mathcal{Y}}_k^{(m+1)} \neq \{\}$, mamy $\dot{\mathcal{Y}}_k^{(\infty)} := \bigcap_{m=k}^{\infty} \dot{\mathcal{Y}}_k^{(m)} \neq \{\}$. Definiujemy model m_k=∞, aby wyprowadzić dowolne $\dot{y}_k^{(\infty)} \in \dot{\mathcal{Y}}_k^{(\infty)}$. To jest najlepszy wynik zgodny z pewnym dowolnym dużym wyborem m_k. Wybór leksykograficznie najmniejszego $\dot{y}_k^{(\infty)} \in \dot{\mathcal{Y}}_k^{(\infty)}$ odpowiadałby dolnej granicy $\underline{\lim}_{m\to\infty} \dot{y}_k^{(m)}$, która zawsze istnieje (dla skończonego Y). Generalnie $\dot{y}_k^{(\infty)} \in \dot{\mathcal{Y}}_k^{(\infty)}$ jest unikalne, tj. $|\dot{\mathcal{Y}}_k^{(\infty)}| = 1$ jeśli istnieje naiwna granica $\lim_{m\to\infty} \dot{y}_k^{(m)}$ k. Należy zauważyć, że granica $\lim_{m\to\infty} V_{km}^*(yx_{<k})$ nie musi istnieć dla tej konstrukcji.

Średnia nagroda i różnicowy zysk. Przyjęcie surowej średniej nagrody $(r_k+...+rm)/(m-k+1)$ i $m\to\infty$ również nie pomaga: rozważ dowolną politykę dla pierwszych k cykli i optymalną politykę dla pozostałych cykli k+1...∞. W środowiskach np. i.i.d. granica istnieje, ale wszystkie te polityki dają tę samą średnią wartość, ponieważ zmiana skończonej liczby wyrazów nie wpływa na nieskończoną średnią. W środowiskach MDP z pojedynczą klasą rekurencyjną można zdefiniować względny lub różnicowy zysk. W bardziej ogólnych środowiskach (którymi jesteśmy zainteresowani) różnicowy zysk może być nieskończony, co jest akceptowalne, ponieważ różnicowe zyski mogą być nadal całkowicie uporządkowane. Głównym problemem jest istnienie różnicowego zysku, tj. czy w ogóle zbiega się dla $m\to\infty$ w IRU{∞} (i nie oscyluje). To jest po prostu stary problem zbieżności w nieco innej formie.

Nieśmiertelni agenci są leniwi. Konstrukcja w akapicie poprzedzającym prowadzi do matematycznie eleganckiego, bezparametrowego modelu AIξ. Niestety, to nie koniec historii. Granica $m_k\to\infty$ może powodować niepożądane rezultaty w modelu AIμ dla specjalnego μ, co może się również zdarzyć w modelu AIξ, niezależnie od tego, jak zdefiniujemy $m_k\to\infty$. Rozważmy agenta, który co √l kolejnych dni pracy może następnie wziąć l dni urlopu. Formalnie rozważmy Y =X =R={0,1}. Wyjście y_k = 0 da nagrodę r_k = 0, a wyjście y_k = 1 da r_k=1 wtedy i tylko wtedy, gdy $\dot{y}_{k-l-\sqrt{l}}\cdots\dot{y}_{k-l}=0$...0 dla pewnego l, tj. agent może osiągnąć l kolejnych pozytywnych nagród, jeśli istniała poprzednia sekwencja o długości co najmniej √l z $y_k = r_k$ = 0. Jeśli czas życia agenta AIμ wynosi m, to wyprowadza on $\dot{y}_k = 0$ w pierwszych s cyklach, a następnie $\dot{y}_k = 1$ dla pozostałych s^2 cykli z s takim, że $s+s^2$ = m. Doprowadzi to do najwyższej możliwej całkowitej nagrody $V_{1m} = s^2 = m + 1/2 - \sqrt{m+1/4}$. Jakakolwiek fragmentacja sekwencji 0 i 1 zredukowałaby V_{1m}, np. naprzemienna praca przez 2 dni i branie 4 dni wolnego dałoby V_{1m}= 2/3m. Dla $m\to\infty$agent AIμ może i będzie opóźniać punkt s przełączenia na $\dot{y}_k = 1$ w nieskończoność i zawsze będzie wyprowadzał 0, co prowadzi do całkowitej nagrody 0, co jest oczywiście najgorszym możliwym zachowaniem. Agent AIξ zbada powyższą regułę po pewnym czasie próbowania y_k=0/1, a następnie zastosuje to samo zachowanie co agent AIμ, ponieważ najprostsze

reguły obejmujące dane z przeszłości dominują w ξ. Dla skończonego m jest to dokładnie to, czego chcemy, ale dla nieskończonego m model AIξ (prawdopodobnie) zawodzi, tak jak model AIμ. Dobra strona jest taka, że nie jest to w szczególności słabość modelu AIξ, ponieważ AIμ również zawodzi. Zła strona jest taka, że $m_k \to \infty$ ma daleko idące konsekwencje, nawet gdy zaczyna się od już bardzo dużego $m_k=m$. Dzieje się tak, ponieważ μ w tym przykładzie jest wysoce nielokalny w czasie, tj. może naruszać jeden z naszych słabych warunków separowalności.

Wnioski. Nie jesteśmy pewni, czy wybór m_k ma marginalne znaczenie, dopóki m_k jest wybierane wystarczająco duże i o niskiej złożoności, na przykład $\tilde{m}_k = 2^{2^{16}}$, lub czy wybór m_k okaże się centralnym tematem dla modelu AIξ lub dla aspektu planowania dowolnego systemu AI w ogólności. Zakładamy, że granica $m_k \to \infty$ dla modelu AIξ skutkuje prawidłowym zachowaniem dla słabo separowalnego μ. Dowód tej hipotezy, jeśli jest prawdziwy, prawdopodobnie dałby interesujące spostrzeżenia.

Perspektywy

Podejście oparte na poradach ekspertów. Rozważaliśmy oczekiwane granice wydajności dla prognoz opartych na wcześniejszych założeniach Solomonoffa. Drugie, podwójne, obecnie bardzo popularne podejście to „predykcja z poradami ekspertów" (PEA) wymyślone przez Littlestone'a i Warmutha oraz Vovka. Podczas gdy PEA działa dobrze w każdym środowisku, ale tylko w odniesieniu do danego zestawu ekspertów, nasz predyktor Λ_ξ konkuruje z każdym innym predyktorem, ale tylko w oczekiwaniu dla środowisk z rozkładem obliczalnym. Wydaje się filozoficznie mniej kompromisowe przyjmowanie założeń dotyczących strategii predykcji niż środowiska, jakkolwiek słabego. Można by zbadać, czy PEA można uogólnić na przypadek aktywnych agentów, co skutkowałoby modelem dualnym do AIXI. Uważamy, że odpowiedź jest negatywna, co po stronie pozytywnej pokazałoby konieczność założenia brzytwy Ockhama i wyjątkowość AIXI. Działania jako zmienne losowe. Można zbadać unikalność wyboru uogólnionego ξ (16) w modelu AIXI. Z pierwotnie wielu alternatyw, które można by wykluczyć, istnieje jedna, która nadal wydaje się możliwa. Zamiast definiować ξ jak w (21), można by traktować działania agenta y również jako zmienne losowe o uniwersalnym rozkładzie, a następnie warunkować ξ na y za pomocą reguły łańcuchowej. Struktura AIXI. Właściwości algebraiczne i struktura AIXI mogłyby zostać zbadane bardziej szczegółowo. Wyciągnęłoby to esencje z AIXI, co ostatecznie mogłoby doprowadzić do aksjomatycznej charakterystyki AIXI. Korzyść jest taka jak w każdym podejściu aksjomatycznym. Jasno pokazałoby założenia, oddzieliłoby esencje od szczegółów technicznych, uprościłoby zrozumienie i, co najważniejsze, poprowadziłoby w znajdowaniu dowodów.

Ograniczone klasy zasad. Rozwój w tej sekcji można by ograniczyć do ograniczonych klas zasad P. Można zdefiniować $V^* = \text{argmax}_{p\in\mathcal{P}} V^p$. Na przykład rozważmy skończoną klasę szybko obliczalnych polityk. W przypadku MDPS, ξ jest szybko obliczalne, a V^p_ξ można (efektywnie) obliczyć za pomocą próbkowania Monte Carlo. Maksymalizacja w skończonej liczbie polityk p∈P wybiera asymptotycznie najlepszą politykę p^ξ z P dla wszystkich (ergodycznych) MDPs [26].

Wnioski

Wszystkie zadania, których rozwiązanie wymaga inteligencji, można naturalnie sformułować jako maksymalizację pewnej oczekiwanej użyteczności w ramach agentów. Podaliśmy jawne wyrażenie (11) takiego agenta teoretyczno-decyzyjnego. Głównym pozostałym problemem jest nieznany rozkład prawdopodobieństwa a priori μ^{AI} środowiska (środowisk). Konwencjonalne algorytmy uczenia się są nieodpowiednie, ponieważ nie mogą obsługiwać dużych (niestrukturyzowanych) przestrzeni stanów

ani nie zbiegają się w teoretycznie minimalnej liczbie cykli, ani nie mogą odpowiednio obsługiwać środowisk niestacjonarnych. Z drugiej strony, uniwersalna półmiara ξ (16), oparta na pomysłach z algorytmicznej teorii informacji, rozwiązuje problem nieznanego rozkładu a priori dla problemów indukcyjnych. Nie jest konieczna żadna jawna procedura uczenia się, ponieważ ξ automatycznie zbiega się do μ. Zunifikowaliśmy teorię uniwersalnej predykcji sekwencji z agentem teoretyczno-decyzyjnym, zastępując nieznaną prawdziwą a priori μ^{AI} odpowiednio uogólnioną uniwersalną półmiarą ξ^{AI}. Przedstawiliśmy mocne argumenty, że wynikowy model AIξ jest uniwersalnie optymalny. Ponadto omówiono możliwe rozwiązania problemu horyzontu. Przedstawiamy szereg klas problemów i opisujemy, w jaki sposób model AIξ może je rozwiązać. Obejmują one predykcję sekwencji, gry strategiczne, minimalizację funkcji, a zwłaszcza sposób, w jaki AIξ uczy się uczyć nadzorowanego. W rozdziale 6 opracowujemy zmodyfikowaną wersję AIXItl ograniczoną czasowo (obliczalną).

Ważne klasy problemów

Aby zapewnić dalsze wsparcie dla uniwersalności i optymalności teorii AIξ, w tej sekcji stosujemy AIξ do szeregu klas problemów. Obejmują one przewidywanie sekwencji, gry strategiczne, minimalizację funkcji, a zwłaszcza sposób, w jaki AIξ uczy się uczyć pod nadzorem. Dla niektórych klas podajemy konkretne przykłady, aby rzucić światło na zakres klasy problemów. Najpierw formułujemy każdą klasę problemów w jej naturalny sposób (gdy znany jest $\mu^{problem}$), a następnie konstruujemy formułę w ramach modelu AIμ i udowadniamy jej równoważność. Następnie rozważamy konsekwencje zastąpienia μ przez ξ. Głównym celem jest zrozumienie, dlaczego i w jaki sposób problemy są rozwiązywane przez AIξ. Podkreślamy tylko szczególne aspekty każdej klasy problemów. Nie badamy każdego aspektu dla każdej klasy problemów. Podsekcje można czytać wybiórczo i nie są one niezbędne do zrozumienia reszty.

Przewidywanie sekwencji (SP)

Wprowadziliśmy model AIξ jako unifikację idei teorii decyzji sekwencyjnych i uniwersalnego rozkładu prawdopodobieństwa. Moglibyśmy oczekiwać, że AIξ będzie zachowywać się identycznie jak SPξ, gdy staniemy przed problemem przewidywania sekwencji, ale sprawy nie są takie proste, jak zobaczymy. Wykorzystanie modelu AIμ do przewidywania sekwencji. Zobaczyliśmy, jak przewidywać sekwencje dla znanego i nieznanego rozkładu a priori μ^{SP}. Tutaj rozważamy sekwencje binarne $z_1z_2z_3...\in IB^\infty$ ze znanym prawdopodobieństwem a priori $\mu^{SP}(z_1z_2z_3...)$. Chcemy pokazać, jak można wykorzystać model AIμ do przewidywania sekwencji. Zobaczymy, że dokonuje on tej samej prognozy co agent SPμ. Dla uproszczenia omawiamy tylko szczególną stratę błędu $l_{xy} = 1-\delta_{xy}$, gdzie δ jest symbolem Kroneckera, zdefiniowanym jako $\delta_{ab}=1$ dla a=b i 0 w przeciwnym razie. Najpierw musimy określić, jak model AIμ powinien być używany do przewidywania sekwencji. Następujący wybór jest naturalny: wyjście systemu y_k jest interpretowane jako przewidywanie dla k-tego bitu z_k rozważanego ciągu. Oznacza to, że yk jest binarne ($y_k\in IB=:Y$). Jako reakcję środowiska, agent otrzymuje nagrodę $r_k=1$, jeśli przewidywanie było poprawne ($y_k=z_k$), lub $r_k=0$, jeśli przewidywanie było błędne ($y_k\neq z_k$). Pytanie brzmi, jaka powinna być obserwacja o_k w następnym cyklu. Jednym z wyborów byłoby poinformowanie agenta o poprawnym k-tym bicie ciągu i ustawienie $o_k=z_k$. Ale ponieważ z nagrody r_k w połączeniu z przewidywaniem y_k można wywnioskować prawdziwy bit $z_k = \delta_{ykrk}$, ta informacja jest zbędna. Nie ma potrzeby tego dodatkowego sprzężenia zwrotnego. Więc ustawiamy $o_k = \varepsilon \in O=\{\varepsilon\}$, mając w ten sposób $x_k \equiv r_k\in R\equiv X =\{0,1\}$. Wydajność agenta nie zmienia się, gdy uwzględniamy te zbędne informacje; komplikuje to jedynie notację. Wcześniejsze prawdopodobieństwo μ^{AI} modelu AIμ wynosi

$$\begin{aligned}\mu^{\mathrm{AI}}(y_1\underline{x}_1...y_k\underline{x}_k) &= \mu^{\mathrm{AI}}(y_1\underline{r}_1...y_k\underline{r}_k)\\ &= \mu^{\mathrm{SP}}(\underline{\delta_{y_1r_1}...\delta_{y_kr_k}})\\ &= \mu^{\mathrm{SP}}(\underline{z_1...z_k}) \qquad (28)\end{aligned}$$

W dalszej części pominiemy indeksy górne μ, ponieważ wynikają one jasno z argumentów μ i μ równych w każdym przypadku. Jest to intuicyjnie jasne i można to formalnie wykazać [19, 30], że maksymalizacja przyszłej nagrody V^{μ}_{km} jest identyczna z chciwym maksymalizowaniem natychmiastowej oczekiwanej nagrody V^{μ}_{kk}. W przypadku przewidywania nie ma kompromisu eksploracja-eksploatacja. Dlatego AIμ działa z

$$\begin{aligned}\dot{y}_k &= \arg\max_{y_k} V^{*\mu}_{kk}(\dot{y}\dot{x}_{<k}y_k)\\ &= \arg\max_{y_k}\sum_{r_k} r_k\cdot\mu^{\mathrm{AI}}(\dot{y}\dot{r}_{<k}y\underline{r}_k) = \arg\max_{z_k}\mu^{\mathrm{SP}}(\dot{z}_1...\dot{z}_{k-1}\underline{z}_k) \qquad (29)\end{aligned}$$

Pierwsze równanie to definicja działania agenta (10) z m_k zastąpionym przez k. W drugim równaniu zastosowaliśmy definicję (9) V_{km}. W ostatnim równaniu zastosowaliśmy (28) i $r_k=\delta_{ykzk}$. Tak więc model AIμ przewiduje, że z_k ma maksymalne μ-prawdopodobieństwo, biorąc pod uwagę $\dot{z}_1...\dot{z}_{k-1}$. Ta prognoza jest niezależna od wyboru m_k. Jest to dokładnie schemat prognozowania predyktora sekwencji SPμ ze znanym priorytecie (ze szczególną stratą błędów). Ponieważ ten model był optymalny, AIμ jest również optymalny, tj. ma minimalną liczbę oczekiwanych błędów (maksymalna μ-oczekiwana nagroda) w porównaniu z dowolnym innym schematem prognozowania sekwencji. Z tego wynika, że wartość $V^{*\mu}_{km}$ musi być ściśle związana z oczekiwanym błędem $E^{\Lambda\mu}_m$ (18). Rzeczywiście, można pokazać, że $V^{*\mu}_{1m}=m-E^{\Lambda\mu}_m$, i podobnie dla ogólnych funkcji strat.

Użycie modelu AIξ do przewidywania sekwencji. Teraz chcemy użyć uniwersalnego modelu AIξ zamiast AIμ do przewidywania sekwencji i spróbować wyprowadzić granice błędów/strat analogicznie do (19). Podobnie jak w przypadku AIμ, wyjście agenta y_k w cyklu k jest interpretowane jako przewidywanie dla k-tego bitu z_k rozważanego ciągu. Nagroda wynosi $r_k=\delta_{ykzk}$ i nie ma innych danych wejściowych $o_k = \varepsilon$. Analizę utrudnia to, że ξ nie jest symetryczne w $y_ir_i\leftrightarrow(1-y_i)(1-r_i)$ i (28) nie zachodzi dla ξ. Z drugiej strony ξ^{AI} zbiega się do μ^{AI} w granicy (23), a (28) powinno zachodzić asymptotycznie dla ξ w pewnym sensie. Oczekujemy więc, że wszystko, co udowodniono dla AIμ, zachodzi w przybliżeniu dla AIξ. Model AIξ powinien zachowywać się podobnie do przewidywania Solomonoffa SPξ. W szczególności oczekujemy granic błędów podobnych do (19). Uczynienie tego rygorystycznym wydaje się trudne. W ostatniej sekcji poczyniono pewne ogólne uwagi. Należy zauważyć, że granice takie jak (25) nie mogą być spełnione ogólnie, ale mogłyby być ważne dla AIξ w środowiskach (pseudo)pasywnych. Tutaj skupiamy się na szczególnym przypadku deterministycznego obliczalnego środowiska, tj. środowisko jest sekwencją $\dot{z}=\dot{z}_1\dot{z}_2...$ z $K(\dot{z}_{1:\infty})<\infty$. Ponadto rozważamy tylko najprostszy model horyzontu m_k=k, tj. maksymalizujemy zachłannie tylko kolejną nagrodę. Jest to wystarczające do przewidywania sekwencji, ponieważ nagroda cyklu k zależy tylko od wyniku y_k, a nie od wcześniejszych decyzji. Ten wybór nie jest w żaden sposób wystarczający i zadowalający dla pełnego modelu AIξ, ponieważ jeden pojedynczy wybór m_k powinien służyć dla wszystkich klas problemów AI. Zatem AIξ powinno umożliwiać dobrą predykcję sekwencji dla pewnego uniwersalnego wyboru m_k, a nie tylko dla m_k=k, co zdecydowanie nie wystarcza w przypadku bardziej skomplikowanych problemów AI. Analiza tego ogólnego przypadku jest wyzwaniem na przyszłość. Dla m_k=k model AIξ (22) z o_i= i $r_k \in\{0,1\}$ redukuje się do

$$\dot{y}_k \;=\; \arg\max_{y_k}\sum_{r_k} r_k\cdot\xi(\dot{y}\dot{r}_{<k}y\underline{r}_k) \;=\; \arg\max_{y_k}\xi(\dot{y}\dot{r}_{<k}y_k\underline{1}). \qquad (30)$$

Reakcja środowiskowa $\dot{r}_k$ jest dana przez $\delta_{\dot{y}_k\dot{z}_k}$ wynosi ona 1 dla poprawnej prognozy $(\dot{y}_k=\dot{z}_k)$ i 0 w przeciwnym wypadku. Można wykazać , że liczba błędnych prognoz $E^{AI\xi}_\infty$ modelu AIξ (30) w tych środowiskach jest ograniczona przez

$$E^{\mathrm{AI}\xi}_\infty \;\overset{\times}{\leq}\; 2^{K(\dot{z}_{1:\infty})} \;<\; \infty \qquad (31)$$

dla obliczalnego deterministycznego łańcucha środowiskowego $\dot{z}_1\dot{z}_2\cdots$ Intuicyjna interpretacja jest taka, że każda błędna prognoza eliminuje co najmniej jeden program p o rozmiarze $l(p)\overset{+}{\leq}K(\dot{z})$. Rozmiar jest mniejszy niż $K(\dot{z})$, ponieważ większe polityki nie mogłyby wprowadzić agenta w błąd co do błędnej prognozy, ponieważ istnieje program o rozmiarze $K(\dot{z})$ dokonujący poprawnej prognozy. Istnieje co najwyżej $2^{K(\dot{z})+O(\hat{1})}$ takich polityk, co ogranicza całkowitą liczbę błędów. Wyprowadziliśmy skończone ograniczenie dla $E^{AI\xi}_\infty$, ale niestety dość słabe w porównaniu do (19). Powodem silnego ograniczenia w przypadku SP było to, że każdy błąd eliminuje połowę programów. Model AIξ nie byłby wystarczający dla realistycznych zastosowań, gdyby ograniczenie (31) było ostre, ale mamy silne przeczucie (ale tylko słabe argumenty), że istnieją lepsze ograniczenia proporcjonalne do $K(\dot{z})$ analogiczne do (19). Obecna technika dowodzenia nie jest wystarczająco silna, aby to osiągnąć. Jednym z argumentów za lepszym ograniczeniem jest formalne podobieństwo między $\mathrm{argmax}_{z_k}\xi(\dot{z}_{<k}\underline{z}_k)$ i (30), drugim jest to, że nie byliśmy w stanie skonstruować przykładowej sekwencji, dla której AIξ powoduje więcej niż $O(K(\dot{z}))$ błędów.

Gry strategiczne (SG)

Wprowadzenie. Gry strategiczne (SG) są bardzo ważną klasą problemów. Teoria gier rozważa proste gry losowe, takie jak ruletka, w połączeniu ze strategią, taką jak backgammon, aż po czysto strategiczne gry, takie jak szachy, warcaby lub go. W rzeczywistości to, co jest objęte teorią gier, jest tak ogólne, że obejmuje nie tylko ogromną różnorodność typów gier, ale może również opisywać polityczne i ekonomiczne zawody i koalicje, darwinizm i wiele innych tematów. Wydaje się, że niemal każdy problem sztucznej inteligencji można sprowadzić do formy gry. Niemniej jednak intencją gry jest to, że kilku graczy wykonuje działania z (częściowymi) obserwowalnymi konsekwencjami. Celem każdego gracza jest maksymalizacja pewnej funkcji użyteczności (np. wygranie gry). Zakłada się, że gracze są racjonalni, biorąc pod uwagę wszystkie posiadane przez nich informacje. Różne cele graczy są zwykle sprzeczne. Jeśli zinterpretujemy system AI jako jednego gracza, a środowisko modeluje drugiego racjonalnego gracza, a środowisko zapewnia wzmacniające sprzężenie zwrotne r_k, zobaczymy, że konfiguracja agent-środowisko spełnia wszystkie kryteria gry. Z drugiej strony modele AI mogą obsługiwać bardziej ogólne sytuacje, ponieważ optymalnie oddziałują ze środowiskiem, nawet jeśli środowisko nie jest racjonalnym graczem o sprzecznych celach. Ściśle konkurencyjne gry strategiczne. W dalszej części ograniczymy się do deterministycznych, ściśle konkurencyjnych gier strategicznych13 z naprzemiennymi ruchami. Gracz 1 wykonuje ruch y_k w rundzie k, po którym następuje ruch ok gracza 2. Tak więc gra z n rundami składa się z sekwencji naprzemiennych ruchów $y_1o_1y_2o_2...y_no_n$. Pod koniec gry w cyklu n gra lub ostateczna sytuacja na planszy jest oceniana za pomocą $V(y_1o_1...y_no_n)$. Gracz 1

próbuje zmaksymalizować V, podczas gdy gracz 2 próbuje zminimalizować V. W najprostszym przypadku V wynosi 1, jeśli gracz 1 wygrał grę, V = −1, jeśli gracz 2 wygrał, a V = 0 w przypadku remisu. Zakładamy stałą długość gry n niezależnie od rzeczywistej sekwencji ruchów. W przypadku gier o zmiennej długości, ale maksymalnej możliwej liczbie ruchów n, moglibyśmy dodać ruchy pozorne i uzupełnić długość do n. Optymalna strategia (równowaga Nasha) obu graczy to strategia minimaksowa:

$$\dot{o}_k = \arg\min_{o_k}\max_{y_{k+1}}\min_{o_{k+1}}...\max_{y_n}\min_{o_n} V(\dot{y}_1\dot{o}_1...\dot{y}_k o_k...y_n o_n), \qquad (32)$$

$$\dot{y}_k = \arg\max_{y_k}\min_{o_k}...\max_{y_n}\min_{o_n} V(\dot{y}_1\dot{o}_1...\dot{y}_{k-1}\dot{o}_{k-1}y_k o_k...y_n o_n). \qquad (33)$$

Należy jednak zauważyć, że strategia minimax jest optymalna tylko wtedy, gdy obaj gracze zachowują się racjonalnie. Jeśli na przykład gracz 2 ma ograniczone możliwości lub popełnia błędy, a gracz 1 jest w stanie je odkryć (poprzez poprzednie ruchy), może wykorzystać te słabości i poprawić swoje wyniki, odchodząc od strategii minimax. Przynajmniej klasyczna teoria gier w równowagach Nasha nie bierze pod uwagę ograniczonej racjonalności, podczas gdy agent AIξ powinien.

Wykorzystanie modelu AIμ do gry. Poniżej demonstrujemy zastosowanie modelu AI do gier. Model AIμ przyjmuje pozycję gracza 1. Środowisko zapewnia ocenę V . W przypadku sytuacji symetrycznej moglibyśmy przyjąć drugi model AIμ jako gracza 2, ale dla uproszczenia przyjmujemy środowisko jako drugiego gracza i zakładamy, że ten gracz środowiskowy zachowuje się zgodnie ze strategią minimaksową (32). Środowisko służy jako doskonały gracz i nauczyciel, choć bardzo prymitywny, ponieważ mówi agentowi na koniec gry tylko, czy wygrał, czy przegrał. Zachowanie minimaksowe gracza 2 można wyrazić za pomocą (deterministycznego) rozkładu prawdopodobieństwa μSG w następujący sposób:

$$\mu^{\mathrm{SG}}(y_1\underline{o}_1...y_n\underline{o}_n) := \begin{cases} 1 & \text{if } o_k = \arg\min_{o'_k}...\max_{y'_n}\min_{o'_n} V(y_1o_1...y_ko'_k...y'_no'_n) \ \ \forall\, k \\ 0 & \text{otherwise} \end{cases}$$

Prawdopodobieństwo, że gracz 2 wykona ruch o_k wynosi $\mu^{\mathrm{SG}}(\dot{y}_1\dot{o}_1...\dot{y}_k\underline{o}_k)$, co wynosi 1 dla $o_k = \dot{o}_k$ zgodnie z definicją w (32) i 0 w przeciwnym wypadku. Oczywiste jest, że system AIμ nie otrzymuje żadnej informacji zwrotnej, tj. $r_1=...=r_{n-1}=0$, aż do końca gry, gdzie powinien otrzymać pozytywną/negatywną/neutralną informację zwrotną w przypadku wygranej/przegranej/remisu, tj. $r_n = V(...)$. Prawdopodobieństwo wstępne środowiska wynosi zatem

$$\mu^{\mathrm{AI}}(y_1\underline{x}_1...y_n\underline{x}_n) = \begin{cases} \mu^{\mathrm{SG}}(y_1\underline{o}_1...y_n\underline{o}_n) & \text{if} \quad r_1...r_{n-1}=0 \\ & \quad \text{and } r_n = V(y_1o_1...y_no_n) \\ 0 & \text{otherwise} \end{cases}, \qquad (35)$$

gdzie $x_i = r_i o_i$. Jeśli otoczenie jest graczem minimaksowym (32) plus prymitywny nauczyciel V , tj. jeśli μ^{AI} jest prawdziwym prawdopodobieństwem a priori, pytanie brzmi teraz: Jakie jest zachowanie $\dot{y}_k^{\mathrm{AI}}$ agenta AIμ? Okazuje się, że jeśli ustawimy $m_k = n$ agent AIμ jest również graczem minimaksowym (33), a zatem optymalnym ($\dot{y}_k^{\mathrm{AI}} = \dot{y}_k^{\mathrm{SG}}$ dla formalnego dowodu). Rozegranie sekwencji gier jest szczególnym przypadkiem faktoryzowalnego μ , z identycznymi czynnikami μ_r dla wszystkich r i równych długości odcinków $n_{r+1}-n_r=n$. Stąd w otoczeniu minimaksowym AIμ zachowuje się jak strategia minimaksowa,

$$\dot{y}_k^{\mathrm{AI}} = \arg\max_{y_k}\min_{o_k}\ldots\max_{y_{(r+1)n}}\min_{o_{(r+1)n}} V(\dot{y}\dot{o}_{rn+1:k-1}\ldots y o_{k:(r+1)n}) \qquad (36)$$

z r takim, że r_n <k ≤ (r+1)n i dla dowolnego wyboru m_k, o ile horyzont h_k≥n.

Używanie modelu AIξ do gry. Przechodząc od konkretnego modelu AIμ, w którym reguły gry są jawnie modelowane w prawdopodobieństwie a priori μ^{AI}, do uniwersalnego modelu AIξ, musimy zapytać, czy te reguły można poznać z przypisanych nagród r_k. Tutaj pojawia się główny powód badania przypadku powtarzanych gier, a nie tylko jednej gry. W przypadku pojedynczej gry istnieje tylko jeden cykl nietrywialnego sprzężenia zwrotnego, a mianowicie koniec gry, który jest zbyt późny, aby był przydatny, chyba że następują kolejne gry. Oczekujemy, że żaden inny schemat uczenia się (bez dodatkowych informacji) nie może nauczyć się gry szybciej niż AIξ, ponieważ μ^{AI} rozkłada się w przypadku gier o stałej długości, tj. μ^{AI} spełnia silny warunek separowalności. W przypadku gry o zmiennej długości splątanie jest również niskie. μ^{AI} nadal powinno być wystarczająco rozdzielne, co pozwoli nam sformułować i udowodnić dobre granice nagrody dla AIξ. Jakościowy argument wygląda następująco:

Ponieważ początkowo AIξ przegrywa wszystkie gry, stara się wyciągać stratę tak długo, jak to możliwe, bez doświadczenia lub nawet wiedzy, co oznacza wygrana. Początkowo AIξ wykona wiele nielegalnych ruchów. Jeśli nielegalne ruchy przerywają grę, co skutkuje (bez opóźnienia) negatywną nagrodą (stratą), AIξ może szybko nauczyć się typowo prostych zasad dotyczących legalnych ruchów, które zazwyczaj stanowią większość zasad; brakuje tylko zasady celu. Po nauczeniu się zasad ruchu, AIξ uczy się (negatywnie nagradzanych) przegrywających pozycji, pozycji prowadzących do przegrywających pozycji itp., więc może próbować wyciągać przegrywające gry. Na przykład w szachach uniknięcie mata przez 20, 30, 40 ruchów przeciwko mistrzowi jest już sporym osiągnięciem. Na tym etapie umiejętności AIξ powinien być w stanie wygrać kilka gier dzięki szczęściu lub spekulować na temat symetrii w grze, że mata szachowa przeciwnika zostanie pozytywnie nagrodzona. Po zapoznaniu się z pełnymi zasadami (ruchy i cel), AIξ od razu uzna, że gra w minimaks jest najlepsza i pokona wszystkich arcymistrzów. Jeśli (skomplikowanej) gry nie można nauczyć się w ten sposób w realistycznej liczbie cykli, należy zapewnić więcej informacji zwrotnych. Można to osiągnąć poprzez pośrednią pomoc w trakcie gry. Środowisko może dawać pozytywne (negatywne) informacje zwrotne dla każdego dobrego (złego) ruchu wykonanego przez agenta. Żądanie, czy ruch ma być oceniany jako dobry, powinno być dostosowane do zdobytego doświadczenia agenta w taki sposób, aby mniej więcej lepsza połowa ruchów była oceniana jako dobra, a druga połowa jako zła, w celu zmaksymalizowania zawartości informacyjnej informacji zwrotnej. W przypadku bardziej skomplikowanych gier, takich jak szachy, z praktycznego punktu widzenia może być konieczne jeszcze więcej informacji zwrotnych. Jednym ze sposobów zwiększenia informacji zwrotnej znacznie poza kilka bitów na cykl jest wyszkolenie agenta poprzez nauczanie go dobrych ruchów. Nazywa się to uczeniem nadzorowanym. Pomimo faktu, że model AIμ ma tylko sprzężenie zwrotne nagrody rk, jest w stanie uczyć się nadzorowanego. Innym sposobem byłoby rozpoczęcie od prostszych gier zawierających pewne aspekty prawdziwej gry i przejście do prawdziwej gry, gdy agent nauczy się prostej gry. Nie oczekuje się żadnych innych trudności podczas przechodzenia od μ do ξ. Ostatecznie ξ^{AI} zbiegnie się do strategii minimaksowej μ^{AI}. W bardziej realistycznym przypadku, gdy środowisko nie jest idealnym graczem minimaksowym, AIξ może wykryć i wykorzystać słabość przeciwnika. Na koniec chcemy skomentować przestrzeń wejścia/wyjścia X/Y modeli AI. W praktycznych zastosowaniach Y prawdopodobnie będzie obejmować również nielegalne ruchy. Jeśli Y jest zbiorem ruchów, np. ramienia robota, agent może przesunąć niewłaściwą figurę lub nawet przewrócić figury. Prostym sposobem radzenia sobie z nielegalnymi ruchami y_k jest interpretowanie ich jako ruchów przegrywających, które kończą grę. Ponadto, jeśli np. wejście x_k jest

obrazem kamery wideo, która wykonuje jeden strzał na ruch, X nie jest zbiorem ruchów środowiska, ale obejmuje zbiór stanów planszy gry. Dyskusja w tej sekcji zajmuje się również tym przypadkiem. Nie ma potrzeby jawnego projektowania przestrzeni wejścia/wyjścia systemu X/Y dla konkretnej gry. Powyższa dyskusja na temat agenta AIξ była raczej nieformalna z następującego powodu: granie (agent SGξ) ma (prawie) taką samą złożoność jak w pełni ogólna AI, a ilościowe wyniki dla agenta AIξ są trudne (ale nie niemożliwe) do uzyskania.

Minimalizacja funkcji (FM)

Zastosowania/przykłady. Istnieje wiele problemów, które można sprowadzić do problemów minimalizacji funkcji (FM). Należy znaleźć minimum funkcji (o wartościach rzeczywistych) f :Y→IR w pewnej dziedzinie Y lub dobre przybliżenie do minimum, zwykle przy ograniczonych zasobach. Jednym z popularnych przykładów jest problem komiwojażera (TSP). Y to zbiór różnych tras między miastami, a f(y) długość trasy y∈Y. Zadanie polega na znalezieniu trasy o minimalnej długości obejmującej wszystkie miasta. Ten problem jest NP-trudny. Uzyskanie dobrych przybliżeń w ograniczonym czasie ma ogromne znaczenie w różnych zastosowaniach. Innym przykładem jest minimalizacja kosztów produkcji (MPC), np. samochodu, przy kilku ograniczeniach. Y to zbiór wszystkich alternatywnych projektów samochodów i metod produkcji zgodnych ze specyfikacjami, a f(y) całkowity koszt alternatywy y ∈Y. Powiązanym przykładem jest znajdowanie materiałów lub (bio)cząsteczek o określonych właściwościach (MAT), np. ciał stałych o minimalnym oporze elektrycznym lub maksymalnie wydajnych modyfikacjach chlorofilu, lub cząsteczek aromatycznych, które smakują jak najbardziej podobnie do truskawek. Możemy również poprosić o ładne obrazy (NPT). Y jest zbiorem wszystkich istniejących lub wyobrażalnych obrazów, a f(y) charakteryzuje, jak bardzo osoba A lubi malować y. Agent powinien przedstawić obrazy, które A lubi. Na razie to są wystarczające przykłady. TSP jest bardzo rygorystyczny z matematycznego punktu widzenia, ponieważ f, tj. algorytm f, jest zwykle znany. W zasadzie minimum można by znaleźć poprzez wyczerpujące wyszukiwanie, gdyby nie ograniczenia zasobów obliczeniowych. W przypadku MPC f można często modelować w sposób niezawodny i wystarczająco dokładny. W przypadku MAT potrzebne są bardzo dokładne modele fizyczne, które mogą być niedostępne lub zbyt trudne do rozwiązania lub wdrożenia. W przypadku NPT mamy tylko osąd osoby A na temat każdego zaprezentowanego obrazu. Funkcji oceny f nie można zaimplementować bez skanowania mózgu A, co nie jest możliwe przy dzisiejszej technologii. Istnieją więc różne ograniczenia, niektóre zależą od aplikacji, którą mamy na myśli. Implementacja f może być niedostępna, f można przetestować tylko przy niektórych argumentach y, a f(y) jest określane przez środowisko.

Chcemy (w przybliżeniu) zminimalizować f przy użyciu jak najmniejszej liczby wywołań funkcji lub odwrotnie, znaleźć jak najbliższe przybliżenie minimum w ustalonej liczbie ocen funkcji. Jeśli f jest dostępne lub może być szybko wywnioskowane przez agenta, a ocena jest szybka, ważniejsze jest zminimalizowanie całkowitego czasu potrzebnego na wyobrażenie sobie nowych kandydatów na minimum próbne plus czas oceny dla f. Ponieważ nie rozważamy aspektów obliczeniowych AIξ do rozdziału 6, koncentrujemy się na pierwszym przypadku, w którym f nie jest dostępne lub dominuje nad wymaganiami obliczeniowymi.

Model zachłanny. Model FM składa się z sekwencji $\dot{y}_1\dot{z}_1\dot{y}_2\dot{z}_2\cdots$ gdzie $\dot{y}_k$ jest próbą agenta FM dla minimum f, a $\dot{z}_k = f(\dot{y}_k)$ jest prawdziwą wartością funkcji zwróconą przez środowisko. Randomizujemy model, zakładając rozkład prawdopodobieństwa μ(f) nad funkcjami. Istnieje kilka powodów, dla których to robimy. Możemy naprawdę nie znać dokładnej funkcji f, jak w przykładzie

NPT, i modelować naszą niepewność za pomocą rozkładu prawdopodobieństwa μ. Co ważniejsze, chcemy przeprowadzić równoległe obliczenia z innymi klasami AI, jak w modelu SPμ, gdzie zawsze zaczynaliśmy od rozkładu prawdopodobieństwa μ, który ostatecznie został zastąpiony przez ξ, aby uzyskać uniwersalną prognozę Solomonoffa SPξ. Chcemy zrobić to samo tutaj. Ponadto przypadek probabilistyczny obejmuje przypadek deterministyczny, wybierając $\mu(f)=\delta_{ff0}$, gdzie f_0 jest prawdziwą funkcją. Ostatnim powodem jest to, że przypadek deterministyczny jest trywialny, gdy μ, a zatem f_0, są znane, ponieważ agent może wewnętrznie (wirtualnie) sprawdzić wszystkie argumenty funkcji i wyprowadzić poprawne minimum od samego początku. Zakładamy, że Y jest przeliczalne, a μ jest miarą dyskretną, np. poprzez branie tylko funkcji obliczalnych. Prawdopodobieństwo, że wartości funkcji $y_1,...,y_n$ wynoszą $z_1,...,z_n$, jest wtedy podane przez

$$\mu^{\mathrm{FM}}(y_1\underline{z}_1...y_n\underline{z}_n) \;:=\; \sum_{f:f(y_i)=z_i\;\forall 1\le i\le n} \mu(f). \qquad (37)$$

Zaczynamy od modelu, który minimalizuje oczekiwanie z_k wartości funkcji f dla następnego wyniku y_k, biorąc pod uwagę poprzednie informacje:

$$\dot{y}_k \;:=\; \arg\min_{y_k}\sum_{z_k} z_k\cdot\mu(\dot{y}_1\dot{z}_1...\dot{y}_{k-1}\dot{z}_{k-1}y_k\underline{z}_k).$$

Ten typ algorytmu chciwego, minimalizującego tylko następne sprzężenie zwrotne, był wystarczający do przewidywania sekwencji (SP) i jest również wystarczający do klasyfikacji (CF, nieopisanej tutaj). Nie jest jednak wystarczający do minimalizacji funkcji, jak pokazuje poniższy przykład. Weźmy f : {0,1}→{1,2,3,4}. Istnieje 16 różnych funkcji, które powinny być równie prawdopodobne, μ(f)= 1/16 . Oczekiwanie funkcji w pierwszym cyklu

$$\langle z_1\rangle \;:=\; \sum_{z_1} z_1\cdot\mu(y_1\underline{z}_1) \;=\; \tfrac{1}{4}\sum_{z_1} z_1 \;=\; \tfrac{1}{4}(1+2+3+4) \;=\; 2.5$$

jest po prostu średnią arytmetyczną możliwych wartości funkcji i jest niezależna od y_1. Dlatego $\dot{y}_1=0,$, jeśli zdefiniujemy argmin, aby przyjąć leksykograficznie pierwsze minimum w niejednoznacznym przypadku, jak tutaj. Załóżmy, że $f_0(0)=2$, gdzie f0 jest prawdziwą funkcją środowiskową, tj. $\dot{z}_1=2.$. Oczekiwanie z_2 wynosi wtedy

$$\langle z_2\rangle \;:=\; \sum_{z_2} z_2\cdot\mu(02y_2\underline{z}_2) \;=\; \begin{cases} 2 & \text{for} \quad y_2=0 \\ 2.5 & \text{for} \quad y_2=1 \end{cases}.$$

Dla y_2=0 agent już wie, że f(0)=2, dla y_2=1 oczekiwanie jest, ponownie, średnią arytmetyczną. Agent ponownie wyprowadzi $\dot{y}_2=0$ ze sprzężeniem zwrotnym $\dot{z}_2=2.$. Będzie to trwało w nieskończoność. Agent nie jest zmotywowany do eksplorowania innych y, ponieważ f(0) jest już mniejsze od oczekiwania f(1). Oczywiste jest, że nie tego chcemy. Model zachłanny zawodzi. Agent powinien być pomysłowy i próbować innych wyjść, gdy da mu się wystarczająco dużo czasu. Ogólnym powodem niepowodzenia podejścia zachłannego jest to, że informacje zawarte w sprzężeniu zwrotnym z_k zależą od wyjścia y_k. Agent FM może aktywnie wpływać na wiedzę, którą otrzymuje ze środowiska, poprzez wybór w y_k. Może być korzystniejsze najpierw zebranie pewnej wiedzy o f poprzez (w sensie zachłannym) nieoptymalny wybór dla y_k, niż natychmiastowe zminimalizowanie oczekiwania z_k. Nieminimalność z_k może zostać nadmiernie skompensowana w dłuższej perspektywie poprzez

wykorzystanie tej wiedzy. W SP otrzymana informacja jest zawsze bieżącym bitem sekwencji, niezależnie od tego, co SP przewiduje dla tego bitu. Dlatego strategia zachłanna w przypadku SP jest już optymalna.

Ogólny model FMμ/ξ. Aby uzyskać użyteczny model, musimy dokładniej zastanowić się nad tym, czego naprawdę chcemy. Czy agent FM powinien wyprowadzić dobre minimum w ostatnim wyjściu w ograniczonej liczbie cykli m, czy średnia wartości $z_1,...,z_m$ powinna być minimalna, czy wystarczy, że tylko jedno z jest tak małe, jak to możliwe? . W dalszej części skupiamy się na minimalizacji średniej lub równoważnie sumie wartości funkcji. Definiujemy model FMμ tak, aby zminimalizować sumę $z_1+...+z_m$. Budowanie średniej μ przez sumowanie po z_i i minimalizowanie względem y_i musi być wykonywane w prawidłowej kolejności chronologicznej. Przy podobnym rozumowaniu, jak w (7) do (11), otrzymujemy

$$\dot{y}_k^{\mathrm{FM}} \;=\; \arg\min_{y_k}\sum_{z_k}...\min_{y_m}\sum_{z_m}(z_1+...+z_m)\cdot\mu(\dot{y}_1\dot{z}_1...\dot{y}_{k-1}\dot{z}_{k-1}y_k\underline{z}_k...y_m\underline{z}_m) \tag{38}$$

Z założenia model FMμ gwarantuje optymalne wyniki w zwykłym sensie, że żaden inny model znający tylko μ nie może dać lepszych wyników. Interesującym przypadkiem (w AI) jest sytuacja, gdy μ jest nieznane. Definiujemy dla tego przypadku model FMξ, zastępując μ(f) pewnym ξ(f), co powinno przypisać wysokie prawdopodobieństwo funkcjom f o niskiej złożoności. Tak więc moglibyśmy zdefiniować $\xi(f)=\Sigma_{q:\forall x[U(qx)=f(x)]}2^{-l(q)}$. Problem z tą definicją polega na tym, że ogólnie rzecz biorąc, nie da się rozstrzygnąć, czy TM q jest implementacją funkcji f. ξ(f) zdefiniowane w ten sposób jest nieobliczalne, a nawet nieaproksymowalne. Ponieważ potrzebujemy tylko ξ analogicznego do lewej strony (37), następująca definicja jest naturalna

$$\xi^{\mathrm{FM}}(y_1\underline{z}_1...y_n\underline{z}_n) \;:=\; \sum_{q:q(y_i)=z_i\ \forall 1\le i\le n} 2^{-l(q)}. \tag{39}$$

ξ^{FM} jest w rzeczywistości równoważne wstawieniu nieobliczalnego ξ(f) do (37). Można pokazać, że ξ^{FM} jest przeliczalną półmiarą i dominuje nad wszystkimi przeliczalnymi rozkładami prawdopodobieństwa postaci (37).

Alternatywnie, moglibyśmy ograniczyć sumę w (39) przez $q(y_1...y_n)= z_1...z_n$ analogicznie do (21), ale te dwie definicje nie są równoważne. Definicja (39) zapewnia symetrię w swoich argumentach i $\xi^{FM}(...y_z...yz'...)=0$ dla $z \neq z'$. Zawiera całą ogólną wiedzę, jaką posiadamy na temat minimalizacji funkcji, podczas gdy (21) nie. Ale ta dodatkowa wiedza ma tylko niską zawartość informacji (złożoność O(1)), więc nie spodziewamy się, że FMξ będzie działać znacznie gorzej, gdy użyjemy (21) zamiast (39). Ale nie ma powodu, aby odchodzić od (39) w tym momencie. Teraz możemy zdefiniować stratę $L^{FM\mu}{}_m$ jako (38) z k=1 i argmin_{y1} zastąpionym przez $\min_{y1}$, a dodatkowo μ zastąpionym przez ξ dla $L^{FM\xi}{}_m$. Oczekujemy $|L^{FM\xi}{}_m - L^{FM\mu}{}_m|$ być ograniczonym w sposób uzasadniający użycie ξ zamiast μ dla obliczalnego μ, tj. obliczalnego f_0 w przypadku deterministycznym. Argumenty są takie same jak dla modelu AIξ. Udowodniono, że FMξ jest pomysłowy w tym sensie, że nigdy nie przestaje szukać minimów, ale przetestuje wszystkie y∈Y, jeśli Y jest skończony (i nieskończony zbiór różnych y, jeśli Y jest nieskończony) dla wystarczająco dużego horyzontu m. Obecnie nie ma rygorystycznych wyników dotyczących jakości zgadywań, ale dla agenta FMμ zgadywania są optymalne z definicji. Jeśli K(μ) dla prawdziwego rozkładu μ jest skończony, oczekujemy, że agent FMξ rozwiąże problem „eksploracji kontra eksploatacja" w sposób uniwersalnie optymalny, ponieważ ξ szybko zbiega do μ.

Wykorzystanie modeli AI do minimalizacji funkcji. Modele AI można wykorzystać do minimalizacji funkcji w następujący sposób. Wyjście y_k cyklu k jest przypuszczeniem minimum f, jak w modelu FM.

Nagroda r_k powinna być wysoka dla małych wartości funkcji $z_k=f(y_k)$. Wybór $r_k=-z_k$ dla nagrody jest naturalny. Tutaj sprzężenie zwrotne nie jest binarne, ale $r_k \in R \subset IR$, gdzie R jest przeliczalnym podzbiorem IR, np. obliczalnymi liczbami rzeczywistymi lub wszystkimi liczbami wymiernymi. Sprzężenie zwrotne ok powinno być wartością funkcji $f(y_k)$. Ponieważ jest to już zapewnione w nagrodach r_k, moglibyśmy ustawić $o_k = \varepsilon$. Dla odmiany i aby zobaczyć, że wybór naprawdę nie ma znaczenia, ustawiamy tutaj $o_k=z_k$. Prawdopodobieństwo wcześniejsze AIμ wynosi

$$\mu^{\mathrm{AI}}(y_1\underline{x}_1...y_n\underline{x}_n) \;=\; \begin{cases} \mu^{\mathrm{FM}}(y_1\underline{z}_1...y_n\underline{z}_n) & \text{for } r_k=-z_k,\; o_k=z_k,\; x_k=r_ko_k \\ 0 & \text{else.} \end{cases} \tag{40}$$

Wstawiając to do (10) z m_k=m można pokazać, że $\dot{y}_k^{\mathrm{AI}} = \dot{y}_k^{\mathrm{FM}}$, gdzie $\dot{y}_k^{\mathrm{FM}}$ zostało zdefiniowane w (38). Dowód jest bardzo prosty, ponieważ model FM ma już dość ogólną strukturę, która jest podobna do pełnego modelu AI. Nie spodziewamy się żadnego problemu w przejściu z FMξ do AIξ. Jedyne, czego model AIξ musi się nauczyć, to ignorować sprzężenia zwrotne o, ponieważ wszystkie informacje są już zawarte w r. To zadanie jest proste, ponieważ każdy cykl dostarcza jeden punkt danych do nauczenia się prostej funkcji.

Uwaga dotycząca TSP. Problem komiwojażera (TSP) wydaje się być trywialny w modelu AIμ, ale nietrywialny w modelu AIξ, ponieważ (38) po prostu implementuje wewnętrzne przeszukiwanie kompletne, ponieważ $\mu(f) = \delta_{ffTSP}$ zawiera wszystkie niezbędne informacje. AIμ od samego początku wyprowadza dokładne minimum f^{TSP}. To „rozwiązanie" jest oczywiście nie do przyjęcia z perspektywy wydajności. Dopóki nie podajemy efektywnego przybliżenia ξ^c ξ, nie wnosimy niczego do rozwiązania TSP przy użyciu $AI\xi^c$. To samo dotyczy każdego innego problemu, w którym f jest obliczalne i łatwo dostępne. Dlatego TSP nie jest (jeszcze) dobrym przykładem, ponieważ wszystko, co zrobiliśmy, to zastąpienie problemu NP-zupełnego nieobliczalnym modelem AIξ lub obliczalnym modelem $AI\xi^c$, dla którego nie powiedzieliśmy jeszcze nic o czasie obliczeń. To po prostu przesada, aby redukować proste problemy do AIξ. TSP jest pod tym względem prostym problemem, dopóki nie rozważymy poważnie modelu $AI\xi^c$. W przypadku innych przykładów, w których f jest niedostępne lub skomplikowane, model $AI\xi^c$ zapewniłby prawdziwe rozwiązanie problemu minimalizacji, ponieważ jawna definicja f nie jest potrzebna dla AIξ i $AI\xi^c$.

Nadzorowane uczenie się na przykładach (EX)

Opracowane modele AI zapewniają ramy dla uczenia się przez wzmacnianie. Środowisko zapewnia sprzężenie zwrotne r, informując agenta o jakości jego ostatniego (lub wcześniejszego) wyniku y; przypisuje nagrodę r do wyniku y. W tym sensie uczenie się przez wzmacnianie jest jawnie zintegrowane z modelami AIμ/ξ. AIμ maksymalizuje prawdziwą oczekiwaną nagrodę, podczas gdy model AIξ jest uniwersalnym, niezależnym od środowiska algorytmem uczenia się przez wzmacnianie. Istnieje inny typ metody uczenia się: Nadzorowane uczenie się przez prezentację przykładów (EX). Wiele problemów nauczonych tą metodą to problemy asocjacyjne następującego typu. Biorąc pod uwagę kilka przykładów $o \in R \subset O$, agent powinien zrekonstruować, z częściowo danego o, brakujące lub uszkodzone części, tj. kompletne o do o takie, że relacja R zawiera o. W wielu przypadkach O składa się z par (z,v), gdzie v jest potencjalnie brakującą częścią. Zastosowania/przykłady. Uczenie się funkcji poprzez prezentowanie par (z, f(z)) i pytanie o wartość funkcji z poprzez prezentowanie (z,?) mieści się w kategorii uczenia nadzorowanego z przykładów, np. f(z) może być etykietą klasy lub kategorią z. Podstawowym przykładem jest uczenie się właściwości obiektów geometrycznych zakodowanych w jakiś sposób. Na przykład, jeśli istnieje 18 różnych obiektów scharakteryzowanych przez ich rozmiar (mały lub duży), ich kolory (czerwony, zielony lub niebieski) i ich kształty (kwadrat, trójkąt lub koło), to

(obiekt,właściwość)∈R, jeśli obiekt posiada tę właściwość. Tutaj R jest relacją, która nie jest wykresem funkcji jednowartościowej. Podczas nauczania dziecka poprzez wskazywanie obiektów i mówienie „to jest drzewo” lub „spójrz, jak zielone” lub „jak piękne”, ustanawia się relację par (obiekt,właściwość) w R. Wskazanie na (być może inne) drzewo później i pytanie „Co to jest?” odpowiada częściowo danej parze (obiekt,?), gdzie brakująca część „?” powinno być uzupełnione przez dziecko mówiące „drzewo”. Ostatnim przykładem są szachy. Widzieliśmy, że w zasadzie szachów można się nauczyć przez uczenie się przez wzmacnianie. W skrajnym przypadku środowisko zapewnia nagrodę r=1 tylko wtedy, gdy agent wygrywa. Szybkość uczenia się jest prawdopodobnie nie do przyjęcia z praktycznego punktu widzenia, ze względu na małą ilość informacji sprzężenia zwrotnego. Bardziej praktyczną metodą nauczania szachów jest przedstawianie przykładowych gier w formie sensownych sekwencji (stan planszy, ruch). Zawierają one informacje o legalnych i dobrych ruchach (ale bez żadnego wyjaśnienia). Po zaprezentowaniu kilku gier nauczyciel może poprosić agenta o wykonanie własnego ruchu, prezentując (stan planszy,?), a następnie ocenić odpowiedź agenta.

Nadzorowane uczenie się z modelem AIμ/ξ. Zdefiniujmy model EX następująco: Środowisko przedstawia dane wejściowe $o_{k-1} = z_k v_k \equiv (z_k,v_k) \in R \cup (Z\times\{?\}) \subset Z\times(Y\cup\{?\}) = O$ dla agenta w cyklu k−1. Oczekuje się, że agent wyprowadzi y_k w następnym cyklu, co jest oceniane z $r_k = 1$, jeśli $(z_k,y_k) \in R$ i 0 w przeciwnym wypadku. Aby uprościć dyskusję, oczekiwane i oceniane jest wyjście y_k nawet wtedy, gdy podane jest v_k(≠?). Aby uzupełnić opis środowiska, należy podać rozkład prawdopodobieństwa $\mu_R(\underline{o_1...o_n})$ przykładów i pytań o_i (zależny od R). Błędne przykłady nie powinny się zdarzyć, tj. μ_R powinno wynosić 0, jeśli $o_i \notin R \cup (Z\times\{?\})$ dla pewnego $1\le i\le n$. Relacje R mogą być również rozłożone prawdopodobieństwem z σ(R). Przykładem prawdopodobieństwa a priori w tym przypadku jest

$$\mu(\underline{o_1...o_n}) = \sum_R \mu_R(\underline{o_1...o_n}) \cdot \sigma(\underline{R}). \qquad (41)$$

Wiedza o wycenie r_k na wyjściu y_k ogranicza możliwe relacje R, zgodne z $R(z_k,y_k)=r_k$, gdzie R(z,y):=1, jeśli (z,y)∈R i 0 w przeciwnym wypadku. Prawdopodobieństwo wcześniejsze dla ciągu wejściowego $x_1...x_n$, jeśli ciąg wyjściowy AIμ to $y_1...y_n$, wynosi zatem

$$\mu^{\mathrm{AI}}(y_1\underline{x_1}...y_n\underline{x_n}) = \sum_{R:\forall 1<i\le n[R(z_i,y_i)=r_i]} \mu_R(\underline{o_1...o_n}) \cdot \sigma(\underline{R})$$

gdzie $x_i = r_i o_i$ i $o_{i-1} = z_i v_i$ z $v_i \in Y\cup\{?\}$. W sekwencji wejścia/wyjścia $y_1x_1y_2x_2...=y_1r_1z_2v_2y_2r_2z_3v_3...$ y_1r_1 są fikcyjne, po czym normalne zachowanie zaczyna się od przykładu (z_2,v_2). Model AIμ jest optymalny dzięki konstrukcji μ^{AI}. Dla obliczalnych wcześniejszych μ_R i σ oczekujemy niemal optymalnego zachowania uniwersalnego modelu AIξ, jeśli μ_R dodatkowo spełnia pewną własność separowalności. Poniżej podajemy pewne uzasadnienie, dlaczego model AIξ uwzględnia informacje o nadzorcy zawarte w przykładach i dlaczego uczy się szybciej niż przez wzmocnienie. Utrzymujemy R stałe i zakładamy, że $\mu_R(o_1...o_n) = \mu_R(o_1)\cdot...\cdot\mu_R(o_n) \ne 0 \Leftrightarrow o_i \in R\cup(Z\times\{?\})$ ∀i, aby uprościć dyskusję. Krótkie kody q wnoszą największy wkład do $\xi^{\mathrm{AI}}(y_1\underline{x_1}...y_n\underline{x_n})$. Ponieważ $o_1...o_n$ jest rozłożone zgodnie z obliczalnym rozkładem prawdopodobieństwa μ_R, krótki kod $o_1...o_n$ dla wystarczająco dużego n jest kodem Huffmana względem rozkładu μ_R. Oczekujemy więc, że μ_R, a zatem R, będą kodowane w dominujących wkładach do ξ^{AI} w jakiś sposób, gdzie przyjęto prawdopodobne założenie, że y na taśmie wejściowej nie mają znaczenia. Zazwyczaj uczy się znacznie więcej niż jednego bitu na cykl, tj. relacja R będzie uczona w n << K(R) cyklach za pomocą odpowiednich przykładów. To kodowanie R w q ewoluuje niezależnie od sprzężeń zwrotnych r. Aby zmaksymalizować sprzężenie zwrotne r_k, agent musi nauczyć

się wyprowadzać y_k z $(z_k,y_k)\in R$. Agent musi wymyślić rozszerzenie programu q do q, które wyodrębni z_k z $o_{k-1}=(z_k,?)$ i wyszuka i wyprowadzi y_k z $(z_k,y_k) \in R$. Ponieważ R jest już zakodowane w q, q' może ponownie wykorzystać to kodowanie R w q. Rozmiar rozszerzenia q' jest zatem rzędu 1. Aby nauczyć się tego q', agent wymaga sprzężenia zwrotnego r z zawartością informacji tylko O(1)=K(q'). Porównajmy to z uczeniem się przez wzmacnianie, gdzie prezentowane są tylko pary $o_{k-1}=(z_k,?)$. Kodowanie R w krótkim kodzie q dla $o_1...o_n$ jest bezużyteczne i dlatego będzie nieobecne. Tylko nagrody r zmuszają agenta do nauczenia się R.

Oczekuje się zatem, że q' będzie miało rozmiar K(R). Zawartość informacyjna w r musi być rzędu K(R). W praktyce często jest tylko bardzo mało rk=1 na początku fazy uczenia się, a zawartość informacyjna w r1...rn jest znacznie mniejsza niż n bitów. Wymagana liczba cykli do nauczenia się R przez wzmocnienie jest zatem co najmniej, ale w wielu przypadkach znacznie większa niż K(R). Chociaż AIξ nigdy nie został zaprojektowany ani nie powiedziano mu, aby uczył się nadzorowany, uczy się, jak korzystać z przykładów od nadzorcy. μ_R i R są uczone na podstawie przykładów; nagrody r nie są konieczne do tego procesu. Pozostałe zadanie nauczenia się, jak uczyć się nadzorowany, jest zatem prostym zadaniem o złożoności O(1), dla którego nagrody r są konieczne.

Inne aspekty inteligencji

W AI opracowano wiele ogólnych idei i metod. W poprzednich podrozdziałach zobaczyliśmy, jak można sformułować kilka klas problemów w ramach AIξ. Ponieważ twierdzimy, że model AIξ jest uniwersalny, chcemy wyjaśnić, które i w jaki sposób inne metody AI są włączane do modelu AIξ, przyglądając się jego strukturze. Niektóre metody są bezpośrednio uwzględniane, podczas gdy inne są lub powinny się pojawiać. Nie twierdzimy, że poniższa lista jest kompletna. Teoria prawdopodobieństwa i teoria użyteczności są sercem modeli AIμ/ξ. Prawdopodobieństwo ξ jest uniwersalnym przekonaniem na temat prawdziwego zachowania środowiskowego μ. Funkcja użyteczności to całkowita oczekiwana nagroda, zwana wartością, która powinna być maksymalizowana. Maksymalizacja oczekiwanej funkcji użyteczności w środowisku probabilistycznym jest zwykle nazywana sekwencyjną teorią decyzji i jest wyraźnie zintegrowana w pełnej ogólności w naszym modelu. W pewnym sensie obejmuje to rozumowanie probabilistyczne (uogólnienie deterministycznego), w którym obiektami rozumowania nie są stwierdzenia prawdziwe i fałszywe, ale przewidywanie zachowania środowiskowego. Wzmocnienie uczenia się jest wyraźnie wbudowane ze względu na nagrody. Nadzorowane uczenie się jest zjawiskiem wyłaniającym się. Algorytmiczna teoria informacji prowadzi nas do użycia ξ jako uniwersalnego oszacowania dla wcześniejszego prawdopodobieństwa μ. Dla horyzontu >1, seria expectimax w (10) i proces wybierania maksymalnych wartości mogą być interpretowane jako abstrakcyjne planowanie. Seria expectimax jest formą wyszukiwania poinformowanego, w przypadku AIμ, i wyszukiwania heurystycznego, dla AIξ, gdzie ξ można interpretować jako heurystykę dla μ. Strategia minimax gry w przypadku AIμ jest również podciągnięta. Model AIξ zbiega się do strategii minimax, jeśli środowisko jest graczem minimax, ale może również wykorzystać graczy środowiskowych o ograniczonej racjonalności. Rozwiązywanie problemów występuje (tylko) w formie maksymalizacji oczekiwanej przyszłej nagrody. Wiedza jest gromadzona przez AIξ i jest przechowywana w pewnej formie, która nie jest określona dalej na taśmie roboczej. Wykorzystywany jest każdy rodzaj informacji w dowolnej reprezentacji na danych wejściowych y. Problem inżynierii wiedzy i reprezentacji pojawia się w postaci sposobu trenowania modelu AIξ. Bardziej praktyczne aspekty, takie jak przetwarzanie języka lub obrazu, muszą być nauczone przez AIξ od podstaw. Inne teorie, takie jak logika rozmyta, teoria możliwości, teoria Dempstera-Shafera itd. są częściowo przestarzałe, a częściowo sprowadzalne do teorii prawdopodobieństwa bayesowskiego. Interpretacja i konsekwencje luki dowodowej g := $1-\sum_{x_k}\xi(yx_{<k}yx_k)>0$ w ξ mogą być podobne do tych w teorii Dempstera-Shafera. Logiczne rozumowanie boolowskie dotyczące świata zewnętrznego odgrywa co najwyżej

wyłaniającą się rolę w modelu AIξ. Inne metody, które wydają się nie być zawarte w modelu AIξ, mogą również być zjawiskami wyłaniającymi się. Model AIξ musi konstruować krótkie kody zachowań środowiskowych, a AIXItl (patrz następna sekcja) musi konstruować krótkie programy działań. Gdybyśmy przeanalizowali i zinterpretowali te programy w realistycznych środowiskach, moglibyśmy znaleźć niektóre z niewymienionych lub nieużywanych lub nowych metod AI w działaniu w tych programach. Jest to jednak w tym momencie czysta spekulacja. Co ważniejsze: próbując uczynić AIξ praktycznie użytecznym, niektóre inne metody AI, takie jak algorytmy genetyczne lub sieci neuronowe, szczególnie do wstępnego/poprzetwarzania I/O, mogą być przydatne. Najważniejszą rzeczą, na którą chcieliśmy zwrócić uwagę, jest to, że model AIξ nie jest pozbawiony żadnej ważnej znanej właściwości inteligencji lub znanej metodologii AI. Brakuje jednak aspektów obliczeniowych, które zostaną omówione w następnej sekcji.

Model AIXI ograniczony czasowo

Do tej pory nie zawracaliśmy sobie głowy nieobliczalnością uniwersalnego rozkładu prawdopodobieństwa ξ. Ponieważ wszystkie uniwersalne modele w tym artykule opierają się na ξ, nie są one skuteczne w tej formie. W tej sekcji opisujemy, w jaki sposób poprzednie modele i wyniki można modyfikować/uogólniać do przypadku ograniczonego czasowo. Rzeczywiście, sytuacja nie jest tak zła, jak mogłaby być. ξ jest wyliczalny, a $\dot{y}_k$ jest nadal aproksymowalny, tj. istnieje algorytm, który wygeneruje sekwencję wyników ostatecznie zbieżnych do dokładnego wyniku $\dot{y}_k$, ale nigdy nie możemy być pewni, czy już go osiągnęliśmy. Poza tym zbieżność jest niezwykle powolna, więc ten typ asymptotycznej obliczalności nie ma bezpośredniego (praktycznego) zastosowania, ale mimo to będzie ważny później. Niech $\tilde{p}$ będzie programem, który oblicza w rozsądnym czasie $\tilde{t}$] na cykl rozsądny inteligentny wynik, tj. ~ $\tilde{p}(\dot{x}_{<k}) = \dot{y}_{1:k}$. Tego rodzaju założenie obliczalności, że komputer ogólnego przeznaczenia o wystarczającej mocy jest w stanie zachowywać się w inteligentny sposób, jest podstawą sztucznej inteligencji, uzasadniając nadzieję na możliwość skonstruowania agentów, którzy ostatecznie osiągną i przewyższą ludzką inteligencję. Nie ma potrzeby omawiania tutaj, co oznacza „rozsądny czas/inteligencja" i „wystarczająca moc". W tej sekcji interesuje nas, czy istnieje obliczalna wersja $\mathrm{AIXI}\tilde{t}$ ~t agenta AIξ, która jest lepsza lub równa dowolnemu p z czasem obliczeń na cykl wynoszącym co najwyżej $\tilde{t}$]. Przez „lepszy" rozumiemy „bardziej inteligentny", więc potrzebujemy relacji porządku dla inteligencji, takiej jak ta w Definicji 5. Najlepszym wynikiem, jaki moglibyśmy wymyślić, byłby $\mathrm{AIXI}\tilde{t}$ z czasem obliczeń ≤ ~t co najmniej tak inteligentny, jak dowolny p z czasem obliczeń $\leq \tilde{t}$. Jeśli AI jest w ogóle możliwa, osiągnęlibyśmy ostateczny cel: skonstruowanie najbardziej inteligentnego algorytmu z czasem obliczeń $\leq \tilde{t}$. Tak jak nie ma uniwersalnej miary w zestawie obliczalnych miar (w czasie $\tilde{t}$), tak samo nie może istnieć taki $\mathrm{AIXI}\tilde{t}$. Możemy realistycznie mieć nadzieję na skonstruowanie agenta $\mathrm{AIXI}\tilde{t}$ o czasie obliczeń c·~t na cykl dla pewnej stałej c. Pomysł polega na uruchomieniu wszystkich programów p o długości $\leq \tilde{l} := l(\tilde{p})$ i czasie ≤~t na cykl i wybraniu najlepszego wyniku. Całkowity czas obliczeń wynosi $c \cdot \tilde{t}$ przy $c = 2^{\tilde{l}}$. Tego rodzaju pomysł „małp piszących na maszynie", z których jedna ostatecznie napisała Szekspira, został zastosowany w różnych formach i kontekstach w teoretycznej informatyce. Realizacja tego najlepszego pomysłu głosowania, w naszym przypadku, nie jest prosta i zostanie przedstawiona w tej sekcji. Powiązanym pomysłem jest oparcie decyzji na większości algorytmów. Ten pomysł „demokratycznego głosowania" został użyty w do przewidywania sekwencji i jest określany jako „ważona większość".

Ograniczone czasowo rozkłady prawdopodobieństwa

W literaturze można znaleźć ograniczone czasowo wersje złożoności Kolmogorowa i ograniczoną czasowo uniwersalną półmiarę . W dalszej części wykorzystujemy i dostosowujemy tę ostatnią, aby zobaczyć, jak daleko zajdziemy. Jednym ze sposobów zdefiniowania ograniczonej czasowo uniwersalnej półmiary chronologicznej jest mieszanka ponad wyliczalnymi półmiarami chronologicznymi obliczalnymi w czasie $\tilde{t}$ i o rozmiarze co najwyżej $\tilde{l}$:

$$\xi^{\tilde{t}\tilde{l}}(y\underline{x}_{1:n}) := \sum_{\rho\,:\,l(\rho)\leq\tilde{l}\,\wedge\,t(\rho)\leq\tilde{t}} 2^{-l(\rho)}\rho(y\underline{x}_{1:n}). \qquad (42)$$

Można pokazać, że $\xi^{\tilde{t}\tilde{l}}$ redukuje się do ξ^{AI} zdefiniowanego w (21) dla $\tilde{t},\tilde{l}\to\infty$. Załóżmy, że prawdziwe prawdopodobieństwo środowiskowe μ^{AI} jest równe lub dostatecznie dokładnie przybliżone przez ρ z $l(\rho)\leq\tilde{l}$ i $t(\rho)\leq\tilde{t}$ i ~l z $\tilde{t}$ i $\tilde{l}$ o rozsądnym rozmiarze. Istnieje kilka problemów AI, które mieszczą się w tej klasie. W minimalizacji funkcji obliczenia f i μ^{FM} są często wykonalne. W wielu przypadkach sekwencje, które powinny zostać przewidziane, można łatwo obliczyć, gdy znana jest μ^{SP}. W problemie klasyfikacji rozkład prawdopodobieństwa μ^{CF}, zgodnie z którym przedstawiono przykłady, jest w wielu przypadkach również elementarny. Ale nie wszystkie problemy AI są tego „łatwego" typu. W przypadku gier strategicznych , samo środowisko jest zazwyczaj wysoce złożonym graczem strategicznym z μ^{SG}, który jest trudny do obliczenia, chociaż można by argumentować, że gracz środowiskowy może mieć również ograniczone możliwości. Łatwo jednak pomyśleć o trudnym do obliczenia fizycznym (probabilistycznym) środowisku, takim jak chemia biocząsteczek. Liczba interesujących zastosowań sprawia, że ta ograniczona klasa problemów AI, z ograniczonym czasem i przestrzenią środowiskiem , $\mu^{\tilde{t}\tilde{l}}$ jest warta zbadania. Indeksy górne rozkładu prawdopodobieństwa z wyjątkiem $\xi^{\tilde{t}\tilde{l}}$ wskazują ich długość i maksymalny czas obliczeń. $\xi^{\tilde{t}\tilde{l}}$ zdefiniowany w (42), z jeszcze nieokreślonym czasem obliczeń, mnoży się przez wszystkie $\mu^{\tilde{t}\tilde{l}}$ tego typu. Stąd model $\mathrm{AI}\xi^{\tilde{t}\tilde{l}}$, w którym używamy $\xi^{\tilde{t}\tilde{l}}$ jako prawdopodobieństwa a priori, jest uniwersalny w stosunku do wszystkich modeli $\mathrm{AI}\mu^{\tilde{t}\tilde{l}}$ w taki sam sposób, w jaki AIξ jest uniwersalny dla AIμ dla wszystkich wyliczalnych półmiar chronologicznych μ. Argmax_{y_k} w (22) wybiera y_k, dla którego $\xi^{\tilde{t}\tilde{l}}$ ma najwyższą oczekiwaną użyteczność V_{kmk}, gdzie $\xi^{\tilde{t}\tilde{l}}$ jest średnią ważoną ponad $\rho^{\tilde{t}\tilde{l}}$; tj. wynik $\dot{y}_k^{\mathrm{AI}\xi^{\tilde{t}\tilde{l}}}$ jest określony przez ważoną większość. Oczekujemy, że $\mathrm{AI}\xi^{\tilde{t}\tilde{l}}$ przewyższy wszystkie (ograniczone) $\mathrm{AI}\rho^{\tilde{t}\tilde{l}}$, analogicznie do przypadku nieograniczonego. W dalszej części analizujemy właściwości obliczalności $\xi^{\tilde{t}\tilde{l}}$ i $\mathrm{AI}\xi^{\tilde{t}\tilde{l}}$, tj. $\dot{y}_k^{\mathrm{AI}\xi^{\tilde{t}\tilde{l}}}$. Aby obliczyć $\xi^{\tilde{t}\tilde{l}}$ zgodnie z definicją (42), musimy wyliczyć wszystkie chronologicznie przeliczalne półmiary $\rho^{\tilde{t}\tilde{l}}$ o długości $\leq\tilde{l}$ i czasie obliczeń $\leq\tilde{t}$. Można to zrobić podobnie do przypadku nieograniczonego. Wszystkie $2^{\tilde{l}}$ przeliczalne funkcje o długości $\leq\tilde{l}$, obliczalne w czasie $\tilde{t}$, muszą zostać przekształcone na chronologiczne rozkłady prawdopodobieństwa. W tym celu należy ocenić każdą funkcję dla |X |·k różnych argumentów. Stąd $\xi^{\tilde{t}\tilde{l}}$ jest obliczalny w czasie $t(\xi^{\tilde{t}\tilde{l}}(y\underline{x}_{1:k}))=O(|\mathcal{X}|\cdot k\cdot 2^{\tilde{l}}\cdot\tilde{t})$. Czas obliczeń $\dot{y}_k^{\mathrm{AI}\xi^{\tilde{t}\tilde{l}}}$ zależy od rozmiaru X, Y i m_k. $\xi^{\tilde{t}\tilde{l}}$ musi zostać obliczony $|Y|^{hk}|X|^{hk}$ razy w (22). Możliwe jest zoptymalizowanie algorytmu i wykonanie obliczeń w czasie

$$t(\dot{y}_k^{\mathrm{AI}\xi^{\tilde{t}\tilde{l}}}) \;=\; O(|\mathcal{Y}|^{h_k}|\mathcal{X}|^{h_k}\cdot 2^{\tilde{l}}\cdot\tilde{t}) \qquad (43)$$

na cykl. Jeśli założymy, że czas obliczeniowy $\mu^{\tilde{t}\tilde{l}}$ wynosi dokładnie $\tilde{t}$ dla wszystkich argumentów, czas siłowy $\bar{t}$ na obliczenie sum i maksimów w (11) wynosi $\bar{t}(\dot{y}_k^{\mathrm{AI}\mu^{\tilde{t}\tilde{l}}}) \geq |\mathcal{Y}|^{h_k}|\mathcal{X}|^{h_k}\cdot\tilde{t}$. Łącząc to z (43), otrzymujemy

$$t(\dot{y}_k^{\mathrm{AI}\xi^{\tilde{t}\tilde{l}}}) \;=\; O(2^{\tilde{l}}\cdot\bar{t}(\dot{y}_k^{\mathrm{AI}\mu^{\tilde{t}\tilde{l}}})).$$

Ten wynik ma proponowaną strukturę, że istnieje uniwersalny agent $\mathrm{AI}\xi^{\tilde{t}\tilde{l}}$ z czasem obliczeniowym $2^{\tilde{l}}$ razy większym niż czas obliczeniowy specjalnego agenta $\mathrm{AI}\mu^{\tilde{t}\tilde{l}}$. Niestety, klasa systemów $\mathrm{AI}\mu^{\tilde{t}\tilde{l}}$ z brutalną oceną $\dot{y}_k$ zgodnie z (11) jest całkowicie nieciekawa z praktycznego punktu widzenia. Na przykład w kontekście szachów powyższy wynik mówi, że $\mathrm{AI}\xi^{\tilde{t}\tilde{l}}$ jest lepszy w czasie $2^{\tilde{l}}\cdot\tilde{t}$ od jakiejkolwiek brutalnej strategii minimaksowej o czasie obliczeniowym $\tilde{t}$. Nawet jeśli czynnik $2^{\tilde{l}}$ w czasie obliczeniowym nie miałby znaczenia, agent $\mathrm{AI}\xi^{\tilde{t}\tilde{l}}$ jest, mimo wszystko, praktycznie bezużyteczny, ponieważ brutalny szachista minimaksowy o rozsądnym czasie $\tilde{t}$ jest bardzo słabym graczem. Należy zauważyć, że w przypadku przewidywania sekwencji binarnych (h_k=1, |Y|=|X |=2) czas obliczeniowy ρ pokrywa się z czasem obliczeniowym $\dot{y}_k^{\mathrm{AI}\rho}$ z dokładnością do czynnika 2. Klasa $\mathrm{AI}\rho^{\tilde{t}\tilde{l}}$ obejmuje wszystkie nieinkrementalne algorytmy przewidywania sekwencji o długości $\leq\tilde{l}$ i czasie obliczeniowym $\leq\tilde{t}/2$. Przez nieinkrementalne rozumiemy, że żadne informacje z poprzednich cykli nie są brane pod uwagę w celu przyspieszenia obliczenia $\dot{y}_k$ bieżącego cyklu. Wady (wspomniane i niewymienione) tego podejścia są naprawiane w następnej podsekcji przez odejście od standardowego sposobu definiowania ograniczonego czasowo ξ jako sumy po funkcjach lub programach.

Idea najlepszego algorytmu głosowania

Ogólny agent to program chronologiczny $p(x_{<k})= y_{1:k}$. Ta forma, jest wystarczająco ogólna, aby objąć dowolny system AI (a także mniej inteligentne systemy). W dalszej części będziemy zainteresowani programami p o długości $\leq\tilde{l}$ i czasie obliczeń $\leq\tilde{t}$ na cykl. Jednym z ważnych punktów w ograniczonym czasowo ustawieniu jest to, że p powinno być przyrostowe, tj. podczas obliczania y_k w cyklu k, informacje z poprzednich cykli zapisane na taśmie roboczej mogą być ponownie wykorzystane. Rzeczywiście, prawdopodobnie nie ma praktycznie żadnego interesującego, nieprzyrostowego systemu AI. W dalszej części konstruujemy politykę p^*, a dokładniej polityki p^*_k dla każdego cyklu k, które przewyższają wszystkie ograniczone czasowo i czasowo systemy AI p. W cyklu k p^*_k uruchamia wszystkie $2^{\tilde{l}}$ programy p i wybiera ten z najlepszym wynikiem y_k. Jest to algorytm typu „najlepszego głosowania”, w porównaniu do algorytmu typu „ważonej większości” z ostatniej podsekcji. Idealnym miernikiem jakości wyników byłaby ξ-oczekiwana przyszła nagroda

$$V_{km}^{p\xi}(\dot{y}\dot{x}_{<k}) \;:=\; \sum_{q\in\dot{Q}_k} 2^{-l(q)} V_{km}^{pq} \quad , \quad V_{km}^{pq} \;:=\; r(x_k^{pq}) + ... + r(x_m^{pq}) \qquad (44)$$

Należy wybrać program p, który maksymalizuje $V^{p\xi}_{kmk}$. Pominęliśmy normalizację N w przeciwieństwie do (24), ponieważ jest ona niezależna od p i nie zmienia relacji porządku, która nas tutaj wyłącznie interesuje. Ponadto bez normalizacji $V_{km}^{*\xi}(\dot{y}\dot{x}_{<k}) := \max_{p\in P} V_{km}^{p\xi}(\dot{y}\dot{x}_{<k})$ jest przeliczalny, co będzie ważne później.

Rozszerzone programy chronologiczne

W formie funkcjonalnej modelu AIξ wygodnie było maksymalizować Vkmk nad wszystkimi $p\in\dot{P}_k$, tj. wszystkimi p zgodnymi z bieżącą historią $\dot{y}\dot{x}_{<k}$. Nie było to ograniczenie, ponieważ dla każdego potencjalnie niespójnego programu p istnieje program $p\in\dot{P}_k$ zgodny z bieżącą historią i identyczny z p dla wszystkich przyszłych cykli ≥k. W przypadku ograniczonego czasowo algorytmu głosowania na najlepsze p* żądanie $p\in\dot{P}_k$ byłoby zbyt restrykcyjne. Aby udowodnić uniwersalność, należy porównać wszystkie algorytmy $2^{\bar{l}}$ w każdym cyklu, a nie tylko te spójne. Niespójny algorytm może stać się najlepszym w późniejszych cyklach. W przypadku programów niespójnych musimy uwzględnić $\dot{y}_k$ w danych wejściowych, tj. $p(\dot{y}\dot{x}_{<k}) = y^p_{1:k}$ gdzie $\dot{y}_i \neq y^p_i$ i jest możliwe. Dla $p\in\dot{P}_k$ nie było to konieczne, ponieważ p zna dane wyjściowe $\dot{y}_k \equiv y^p_k$ w tym przypadku. r^{pq}_i w definicji V_{km} to nagrody pojawiające się w sekwencji wejścia/wyjścia, zaczynając od $\dot{y}\dot{x}_{<k}$ (pojawiającej się z p*), a następnie kontynuowane przez zastosowanie p i q z $\dot{y}_i := y^p_i$ i dla i ≥k. Innym problemem jest to, że potrzebujemy V_{kmk}, aby wybrać najlepszą strategię, ale niestety V_{kmk} jest nieobliczalny. Rzeczywiście, struktura definicji V_{kmk} jest bardzo podobna do struktury $\dot{y}_k$, stąd podejście siłowe do aproksymacji V_{kmk} wymaga zbyt dużo czasu obliczeniowego, jak w przypadku $\dot{y}_k$. Rozwiązujemy ten problem w podobny sposób, uzupełniając każde p programem, który szacuje V_{kmk} przez w^p_k w czasie $\tilde{t}$. Łączymy obliczenia y^p_k i w^p_k i rozszerzamy pojęcie programu chronologicznego raz jeszcze na

$$p(\dot{y}\dot{x}_{<k}) \;=\; w_1^p y_1^p ... w_k^p y_k^p, \qquad (45)$$

w porządku chronologicznym $w_1^p y_1^p \dot{y}_1 \dot{x}_1 w_2^p y_2^p \dot{y}_2 \dot{x}_2 ....$

Prawidłowe przybliżenia

Polityka p może sugerować dowolny wynik y^p_k, ale nie wolno jej oceniać dowolnie wysokim w^p_k, jeśli chcemy, aby w^p_k było wiarygodnym kryterium wyboru najlepszego p. Żądamy, aby żadna polityka nie mogła twierdzić, że jest lepsza, niż jest w rzeczywistości. Definiujemy (logiczny) predykat VA(p) zwany poprawnym przybliżeniem, który jest prawdziwy wtedy i tylko wtedy, gdy p zawsze spełnia $w^p_k \leq V^{p\xi}_{kmk}$, tj. nigdy nie przecenia siebie.

$$\mathrm{VA}(p) \;\equiv\; [\forall k \forall w_1^p y_1^p \dot{y}_1 \dot{x}_1 ... w_k^p y_k^p : p(\dot{y}\dot{x}_{<k}) = w_1^p y_1^p ... w_k^p y_k^p \Rightarrow w_k^p \leq V_{km_k}^{p\xi}(\dot{y}\dot{x}_{<k})] \qquad (46)$$

W dalszej części ograniczamy naszą uwagę do programów p, dla których VA(p) można udowodnić w pewnym formalnym systemie aksjomatycznym. Bardzo ważnym punktem jest to, że $V^{*\xi}_{kmk}$ jest przeliczalne. Zapewnia to istnienie sekwencji programów $p_1,p_2,p_3,...$ dla których VA(p_i) można udowodnić i $\lim_{i\to\infty} w^{pi}_k = V^{*\xi}_{kmk}$ dla wszystkich k i wszystkich sekwencji wejścia/wyjścia. p_i można zdefiniować jako naiwny (niezatrzymujący) schemat przybliżenia (przez enumerację) $V^{*\xi}_{kmk}$ zakończony po i krokach czasowych i wykorzystujący przybliżenie uzyskane dotychczas dla w^{pi}_k wraz z odpowiadającym mu wyjściem y^{pi}_k . Konwergencja $w_k^{p_i} \xrightarrow{i\to\infty} V^{*\xi}_{km_k}$ zapewnia, że $V^{*\xi}_{kmk}$, które uznaliśmy za uniwersalnie optymalną wartość, może być aproksymowane przez p z udowodnionym VA(p) dowolnie dobrze, gdy da się wystarczająco dużo czasu. Aproksymacja nie jest jednostajna w k, ale nie ma to znaczenia, ponieważ wybrane p może się zmieniać z cyklu na cykl. Inną możliwością byłoby rozważenie tylko tych p, które sprawdzają $w^p_k \le V^{p\xi}_{kmk}$ online w każdym cyklu, zamiast wstępnego sprawdzenia VA(p), albo poprzez skonstruowanie dowodu (na taśmie roboczej) dla tego szczególnego przypadku, albo $w^p_k \le V^{p\xi}_{kmk}$ jest już oczywiste dzięki konstrukcji w^p_k. W przypadkach, gdy p nie może zagwarantować $w^p_k \le V^{p\xi}_{kmk}$ ustawia $w_k = 0$ i stąd trywialnie spełnia $w^p_k \le V^{p\xi}_{kmk}$. Z drugiej strony, dla tych p nie stanowi problemu udowodnienie VA(p), ponieważ wystarczy przeanalizować wewnętrzną strukturę p i rozpoznać, że p wykazuje ważność wewnętrznie, cykl po cyklu, co jest łatwe z założenia na p. Sprawdzanie cykl po cyklu jest zatem szczególnym przypadkiem wstępnego dowodu VA(p).

Efektywna relacja porządku inteligencji

Wprowadziliśmy relację porządku $\succcurlyeq$ inteligencji w systemach AI, opartą na oczekiwanej nagrodzie $V^{p\xi}_{kmk}$. W dalszej części potrzebujemy relacji porządku $\succcurlyeq^c$ opartej na żądanej nagrodzie w^p_k, którą można interpretować jako przybliżenie do .

Definicja 7 (Efektywna relacja porządku inteligencji). Nazywamy p efektywnie bardziej lub równie inteligentnym niż p , jeśli

$$p \succeq^c p' :\Leftrightarrow \forall k \forall \dot{y}\dot{x}_{<k} \exists w_{1:n} w'_{1:n} :$$
$$p(\dot{y}\dot{x}_{<k}) = w_1 * ... w_k * \ \wedge p'(\dot{y}\dot{x}_{<k}) = w'_1 * ... w'_k * \ \wedge w_k \geq w'_k,$$

tj. jeśli p zawsze rości sobie prawo do wyższej nagrody szacunkowej w niż p’

. Relacja $\succcurlyeq^c$ jest współprzeliczalną relacją częściowego porządku w rozszerzonych programach chronologicznych. Ograniczona do poprawnych przybliżeń porządkuje ona polityki względem jakości ich wyników i ich zdolności do uzasadniania ich wyników z wysokim w_k.

Uniwersalny agent AIXItl ograniczony czasowo

W dalszej części opisujemy algorytm p^* leżący u podstaw uniwersalnego agenta $\mathrm{AIXI}t\tilde{l}$ ograniczonego czasowo. Opiera się on zasadniczo na wyborze najlepszych algorytmów p^*_k z czasu $\tilde{t}$ i długości $\tilde{l}$ ograniczonych p, dla których istnieje dowód VA(p) o długości $\le l_p$.

1. Utwórz wszystkie ciągi binarne o długości l_p i zinterpretuj każdy z nich jako kodowanie dowodu matematycznego w tym samym formalnym systemie logicznym, w którym sformułowano VA(·). Weź te ciągi, które są dowodami VA(p) dla pewnego p i zachowaj odpowiadające im programy p.

2. Wyeliminuj wszystkie p o długości $> \tilde{l}$.

3. Zmodyfikuj zachowanie wszystkich zachowanych p w każdym cyklu k w następujący sposób: Nic się nie zmienia, jeśli p wyprowadza pewne $w^p_k y^p_k$ w ciągu $\tilde{t}$ kroków czasowych. W przeciwnym wypadku zatrzymaj p i zapisz $w_k=0$ i dowolne y_k na taśmie wyjściowej p. Niech P będzie zbiorem wszystkich tych zmodyfikowanych programów.

4. Rozpocznij pierwszy cykl: k :=1.

5. Uruchom każdy p∈P na rozszerzonym wejściu $\dot{y}\dot{x}_{<k}$, gdzie wszystkie wyjścia są przekierowywane na jakąś taśmę pomocniczą: $p(\dot{y}\dot{x}_{<k}) = w^p_1 y^p_1 ... w^p_k y^p_k$. Ten krok jest wykonywany przyrostowo przez dodanie $\dot{y}\dot{x}_{k-1}$ dla k>1 do taśmy wejściowej i kontynuowanie obliczeń z poprzedniego cyklu.

6. Wybierz program p z najwyższą zgłoszoną nagrodą w^p_k: $p^*_k := \text{argmax}_p w^p_k$.

7. Zapisz $\dot{y}_k := y^{p^*_k}_k$ na taśmie wyjściowej.

8. Odbierz wejście $\dot{x}_k$ ze środowiska.

9. Rozpocznij następny cykl: k :=k+1, przejdź do kroku 5.

Łatwo zauważyć, że obowiązuje następujące twierdzenie.

Twierdzenie 6 (Optymalność AIXItl). Niech p będzie dowolnym rozszerzonym programem chronologicznym (inkrementalnym) podobnym do (45) o długości $l(p) \leq \tilde{l}$ i czasie obliczeń na cykl $t(p) \leq \tilde{t}$, dla którego istnieje dowód VA(p) zdefiniowany w (46) o długości $\leq l_P$. Algorytm p^* skonstruowany w ostatnim akapicie, który zależy od $\tilde{l}$, $\tilde{t}$ i l_P, ale nie od p, jest efektywnie bardziej lub równie inteligentny, zgodnie z $\succcurlyeq^c$ (patrz Definicja 7), niż jakikolwiek taki p. Rozmiar p^* wynosi $l(p^*) = O(\log(\tilde{l} \cdot \tilde{t} \cdot l_P))$, czas konfiguracji wynosi $ts_{etup}(p^*)=O(l^2_P \cdot 2^{l_P})$, a czas obliczeń na cykl wynosi $t_{cycle}(p^*) = O(2^{\tilde{l}} \cdot \tilde{t})$.

Mówiąc ogólnie, twierdzenie mówi, że jeśli istnieje obliczalne rozwiązanie niektórych lub wszystkich problemów AI, jawnie skonstruowany algorytm $p*$ jest takim rozwiązaniem. Chociaż twierdzenie to jest dość ogólne, istnieją pewne ograniczenia i otwarte pytania, które omówimy w następnej podsekcji. Konstrukcja algorytmu p^* wymaga specyfikacji formalnego systemu logicznego (∀, λ, y_i, c_i, f_i, Ri, →, ∧, =,...), aksjomatów i reguł wnioskowania. Dowód to ciąg formuł, gdzie każda formuła jest albo aksjomatem, albo wywnioskowana z poprzednich formuł w ciągu poprzez zastosowanie reguł wnioskowania. Szczegóły można znaleźć w powiązanej konstrukcji lub w dowolnym podręczniku dotyczącym logiki lub teorii dowodu. Musimy tylko wiedzieć, że dowodzenie i maszyny Turinga można sformalizować. Czas przygotowania w twierdzeniu to po prostu czas potrzebny na sprawdzenie dowodów 2lP, z których każdy wymaga czasu $O(l^2_P)$.

Ograniczenia i pytania otwarte

• Formalnie, całkowity czas obliczeniowy p^* dla cykli 1...k wzrasta liniowo z k, tj. jest rzędu O(k) ze współczynnikiem $2^{\tilde{l}} \cdot \tilde{t}$. Nieracjonalnie duży czynnik $2^{\tilde{l}}$ jest dobrze znaną wadą najlepszych/demokratycznych modeli głosowania i zostanie przyjęty bez dalszych komentarzy, podczas gdy czynnik $\tilde{t}$ można założyć, że ma rozsądny rozmiar. Jeśli nie weźmiemy granicy k→∞, ale

rozważymy rozsądne k, praktyczne znaczenie ograniczenia czasowego p^* jest nieco ograniczone ze względu na dodatkową stałą addytywną $O(l^2_P \cdot 2^{l^P})$. Jest ona znacznie większa niż $k \cdot 2^l \cdot \tilde{t}$, ponieważ typowo $l_P \gg l(\mathrm{VA}(p)) \geq l(p) \equiv \tilde{l}$.

• p^* jest lepsze tylko od tych p, które uzasadniają swoje wyniki (przez duże w^p_k). Możliwe, że istnieje p, które produkują dobre wyniki y^p_k w rozsądnym czasie, ale uzasadnienie ich wyników przez wystarczająco wysokie w^p_k zajmuje nierozsądnie dużo czasu. Nie uważamy, aby (od pewnego poziomu złożoności w górę) istniały polityki, w których proces konstruowania dobrego wyniku jest całkowicie oddzielony od jakiegoś rodzaju procesu uzasadniania. Jednak to uzasadnienie może nie być przetłumaczalne (przynajmniej w rozsądnym czasie) na rozsądny szacunek $V^{p\xi}_{kmk}$.

• (Niespójne) programy p muszą być w stanie kontynuować strategie rozpoczęte przez inne polityki. Może się zdarzyć, że polityka p kieruje otoczeniem w kierunku, w którym p jest wyspecjalizowane. Polityka „obca" może być w stanie przemieścić p tylko między luźno powiązanymi epizodami. Prawdopodobnie nie ma problemu dla faktoryzowalnego μ. Pomyśl o grze w szachy, w której zazwyczaj bardzo trudno jest kontynuować grę lub strategię innego gracza. Kiedy gra się kończy, zazwyczaj korzystne jest zastąpienie gracza lepszym w następnej grze. Może również nie być problemu dla wystarczająco rozdzielnego μ.

• Mogą istnieć (efektywne) poprawne przybliżenia p, dla których VA(p) jest prawdziwe, ale nieudowodnialne, lub dla których istnieje tylko bardzo długi (>lP) dowód

Uwagi

• Pomysł sugerowania wyników i uzasadniania ich poprzez udowodnienie granic nagrody implementuje jeden aspekt ludzkiego myślenia. Istnieje kilka możliwych reakcji na dane wejściowe. Każda reakcja może mieć daleko idące konsekwencje. W ograniczonym czasie próbuje się oszacować konsekwencje tak dobrze, jak to możliwe. Na koniec każda reakcja jest wyceniana i wybierana jest najlepsza. To, co jest gorsze od ludzkiego myślenia, to to, że oszacowania w^p_k muszą być rygorystycznie udowodnione, a dowody są konstruowane przez ślepe wyczerpujące wyszukiwanie, ponadto, że wszystkie zachowania p o długości $\leq \tilde{l}$ są sprawdzane. Jest to gorsze „tylko" w sensie niezbędnego czasu obliczeń, ale nie w sensie jakości wyników.

• W praktycznych zastosowaniach często zdarzają się przypadki krótkich i wolnych programów p wykonujących pewne zadanie T, np. obliczenie cyfr π, dla których istnieją również długie, ale szybkie programy pl. Jeśli nie jest zbyt trudno udowodnić, że ten długi program jest równoważny krótkiemu, to można udowodnić $K^{t(p_l)}(T) \stackrel{+}{\leq} l(p_s)$ przy czym K^t jest ograniczoną czasowo złożonością Kołmogorowa. Podobnie, metoda dowodzenia ograniczeń w_k dla V_{kmk} może dać wysokie dolne ograniczenia bez jawnego wykonywania tych krótkich i wolnych programów, które głównie przyczyniają się do V_{kmk} .

• Łączenie wszystkich programów o ograniczonej długości i czasie jest dobrze znanym elementarnym pomysłem (np. małpy piszące na maszynie). Kluczową częścią, która została tutaj opracowana, jest kryterium wyboru najinteligentniejszego agenta.

• Konstrukcja $\mathrm{AIXI}\tilde{t}\tilde{l}$ i wyliczalność V_{kmk} zapewniają dowolne bliskie przybliżenia V_{kmk} , stąd oczekujemy, że zachowanie $\mathrm{AIXI}\tilde{t}\tilde{l}$ zbiega się do zachowania AIξ w granicy $\tilde{t}, \tilde{l}, l_P \to \infty$, w pewnym sensie.

• W zależności od tego, co wiesz lub zakładasz, że program p o rozmiarze $\tilde{l}$ i czasie obliczeniowym na cykl $\tilde{t}$ jest w stanie osiągnąć, obliczalny model $\mathrm{AIXI}\tilde{t}\tilde{l}$ będzie miał takie same możliwości. W przypadku najsilniejszego założenia istnienia maszyny Turinga, która przewyższa inteligencję człowieka, $\mathrm{AIXI}\tilde{t}\tilde{l}$ również to zrobi, w tym samym przedziale czasowym aż do (niestety bardzo dużego) stałego czynnika.

Dyskusja

Ta sekcja omawia to, co zostało osiągnięte i omawia niektóre w inny sposób niewspomniane tematy o ogólnym zainteresowaniu. Zwracamy uwagę na różne tematy, w tym współbieżne działania i percepcje, wybór przestrzeni I/O, przetwarzanie zaszyfrowanych informacji i osobliwości agentów ucieleśniających śmiertelników. Kontynuujemy spojrzenie na dalsze badania. Ponieważ wiele pomysłów zostało już przedstawionych w różnych sekcjach, koncentrujemy się na nietechnicznych otwartych kwestiach o ogólnym znaczeniu, w tym optymalności, zmniejszaniu skali, implementacji, aproksymacji, elegancji, dodatkowej wiedzy i szkoleniu AIXI(tl). Dołączamy również kilka (osobistych) uwag na temat fizyki nieobliczalnej, liczby mądrości Ω i świadomości. Jak należy, rozdział kończy się wnioskami.

Uwagi ogólne

Teoria gier. W teorii gier często chce się modelować sytuację jednoczesnych działań, podczas gdy modele AIξ mają szeregowe wejście/wyjście. Jednoczesność można symulować, powstrzymując środowisko od bieżącego wyjścia agenta y_k, dopóki agent nie otrzyma x_k. Formalnie oznacza to, że $\mu(yx_{<k}y\underline{x}_k)$ jest niezależne od ostatniego wyjścia y_k. Agent AIξ jest już typu jednoczesnego w abstrakcyjnym widoku, jeśli zachowanie p jest interpretowane jako działanie. W tym sensie AIXI jest działaniem p^*, które maksymalizuje funkcję użyteczności (nagrodę), przy założeniu, że środowisko działa zgodnie z ξ. Sytuacja ta różni się od teorii gier, ponieważ środowisko ξ nie jest drugim „graczem”, który próbuje zoptymalizować swoją własną użyteczność . Przestrzenie wejścia/wyjścia. W różnych przykładach wybraliśmy różnie wyspecjalizowane przestrzenie wejściowe i wyjściowe X i Y. Powinno być jasne, że w zasadzie jest to niepotrzebne, ponieważ wystarczająco duże przestrzenie X i Y (np. zbiór ciągów o długości 2^{32}) zaspokajają wszelkie potrzeby i zawsze mogą być zredukowane metodą Turinga do konkretnej prezentacji potrzebnej wewnętrznie samemu agentowi AIXI. Ale jest jasne, że użycie ogólnego interfejsu, takiego jak kamera i monitor do nauki gry w kółko i krzyżyk, na przykład, dodaje zadanie nauki widzenia i rysowania.

Jak AIXI(tl) radzi sobie z zaszyfrowanymi informacjami. Rozważmy zadanie odszyfrowania wiadomości, która została zaszyfrowana przez szyfrator klucza publicznego, taki jak RSA. Wiadomość m jest szyfrowana przy użyciu iloczynu n dwóch dużych liczb pierwszych p_1 i p_2, co daje zaszyfrowaną wiadomość c=RSA(m|n). RSA to prosty algorytm o rozmiarze O(1). Jeśli AIXI otrzyma klucz publiczny n i zaszyfrowaną wiadomość c, aby odtworzyć oryginalną wiadomość m, musi jedynie „nauczyć się” funkcji $\mathrm{RSA}^{-1}(c|n) := \overline{\mathrm{RSA}}(c|p_1,p_2) = m$. RSA^{-1} można opisać na długości O(1), ponieważ $\overline{\mathrm{RSA}}$ jest O(1), a p_1 i p_2 można odtworzyć z n. Do nauczenia się O(1) bitów potrzeba jedynie bardzo niewielu informacji. W tym sensie odszyfrowanie jest łatwe dla AIXI (podobnie jak TSP). Problem polega na tym, że podczas gdy RSA jest wydajny, RSA^{-1} jest niezwykle wolnym algorytmem, ponieważ musi znaleźć czynniki pierwsze z klucza publicznego. Ale zauważ, że w AIXI nie mówimy o czasie obliczeń, mówimy jedynie o wydajności informacji (nauka w najmniejszej liczbie cykli interakcji). Jednym z kluczowych spostrzeżeń w tym artykule, które umożliwiły elegancką teorię AI, było to oddzielenie wydajności danych od wydajności czasu obliczeniowego. Oczywiście w świecie rzeczywistym czas obliczeniowy ma

znaczenie, więc wymyśliliśmy AIXItl. AIXItl może wykonać każde zadanie tak samo dobrze, jak najlepszy agent o długości l i czasie t, z wyjątkiem czynnika czasu 2^l i ogromnego czasu przesunięcia. Żaden praktyczny czas przesunięcia nie jest wystarczający, aby znaleźć czynniki n, ale teoretycznie wystarczający czas przesunięcia pozwala również AIXItl (raz na zawsze) znaleźć faktoryzację, a następnie odszyfrowanie jest oczywiście łatwe.

Śmiertelni ucieleśnieni agenci. Przykłady podane w tym artykule, to głównie bezcielesni agenci: predyktory, hazardziści, optymalizatorzy, uczący się. Istnieją pewne osobliwości w przypadku autonomicznych ucieleśnionych robotów z uczeniem się przez wzmacnianie w rzeczywistych środowiskach. Nadal możemy nagradzać robota w zależności od tego, jak dobrze rozwiązuje zadanie, które chcemy, aby wykonał. Minimalnym wymogiem jest prawidłowe działanie sprzętu robota. Jeśli robot zacznie działać nieprawidłowo, jego możliwości ulegną pogorszeniu, co spowoduje obniżenie nagrody. Tak więc, w celu maksymalizacji nagrody, robot będzie również utrzymywał się. Problem polega na tym, że niektóre części ulegną awarii dość szybko, jeśli nie zostaną wykonane żadne odpowiednie czynności, np. rozładowane baterie, jeśli nie zostaną naładowane na czas. Co gorsza, robot może działać idealnie, dopóki bateria nie będzie prawie pusta, a następnie nagle przestać działać (śmierć), co spowoduje zerową nagrodę od tego momentu. Jest zbyt mało czasu, aby nauczyć się, jak utrzymywać się, zanim będzie za późno. Autonomiczny ucieleśniony robot nie może zacząć od zera, ale musi mieć pewne podstawowe wbudowane zdolności (które mogą wcale nie być takie podstawowe), które pozwolą mu przynajmniej przetrwać. Zwierzęta przeżywają dzięki odruchom, wrodzonemu zachowaniu, wewnętrznej nagrodzie związanej ze stanem ich organów i środowisku opiekuńczemu w dzieciństwie. Różne gatunki kładą nacisk na różne aspekty. U zwierząt niższych kładzie się nacisk na odruchy i wrodzone zachowania w porównaniu z latami bezpiecznego dzieciństwa u ludzi. Ta sama różnorodność rozwiązań jest dostępna do konstruowania autonomicznych robotów (których nie będziemy tutaj szczegółowo omawiać). Inny problem związany, ale prawdopodobnie nieograniczający się do ucieleśnionych agentów, zwłaszcza jeśli są nagradzani przez ludzi, jest następujący: Wystarczająco inteligentni agenci mogą zwiększać swoje nagrody poprzez psychologiczną manipulację swoimi ludzkimi „nauczycielami" lub grożenie im. Jest to ogólny problem socjologiczny, który spowoduje udana sztuczna inteligencja, która nie ma nic wspólnego z AIXI. Każda inteligencja wyższa od ludzkiej jest zdolna do manipulowania tą ostatnią. W przypadku braku manipulacyjnych ludzi, np. gdzie struktura nagrody służy funkcji przetrwania, AIXI może bezpośrednio włamać się do sprzężenia zwrotnego nagrody. Ponieważ jest mało prawdopodobne, aby zwiększyło to jego długoterminowe przetrwanie, AIXI prawdopodobnie będzie opierać się tego rodzaju manipulacjom (podobnie jak większość ludzi nie bierze twardych narkotyków z powodu ich długoterminowych katastrofalnych konsekwencji).

Perspektywy i pytania otwarte

Wiele pomysłów na dalsze badania zostało już przedstawionych w różnych sekcjach artykułu. Ta perspektywa zawiera jedynie nietechniczne pytania otwarte dotyczące AIXI(tl) o ogólnym znaczeniu. Ograniczenia wartości. Rygorystyczne dowody nieasymptotycznych ograniczeń wartości dla AIξ stanowią główne wyzwanie teoretyczne – ogólne, a także ściślejsze ograniczenia dla specjalnych środowisk μ, np. do szybkiego mieszania mdps i/lub innych kryteriów wydajności, muszą zostać znalezione i udowodnione. Chociaż nie jest to konieczne z praktycznego punktu widzenia, badanie klas ciągłych M, klas polityki ograniczonej i/lub nieskończonych Y, X i m może prowadzić do przydatnych spostrzeżeń. Skalowanie AIXI w dół. Bezpośrednia implementacja modelu AIXItl jest w najlepszym przypadku możliwa w środowiskach o małej skali (zabawkowych) ze względu na duży czynnik 2l w czasie obliczeń. Istnieją jednak inne zastosowania teorii AIXI. W kilku przykładach widzieliśmy, jak zintegrować klasy problemów z modelem AIXI. Odwrotnie, można zmniejszyć skalę modelu AIξ,

używając bardziej ograniczonych form ξ. Można to zrobić w ten sam sposób, w jaki zmniejszono skalę teorii indukcji uniwersalnej z wieloma spostrzeżeniami do zasady minimalnej długości opisu lub do dziedziny automatów skończonych . Model AIXI może podobnie służyć jako supermodel lub jako sama definicja (uniwersalnej nieobciążonej) inteligencji, z której można by wyprowadzić wyspecjalizowane modele. Implementacja i aproksymacja. Przy rozsądnym czasie obliczeń model AIXI byłby rozwiązaniem AI (zobacz następny punkt, jeśli się nie zgadzasz). Model AIXItl był pierwszym krokiem, ale wyeliminowanie czynnika 2^l bez rezygnacji z uniwersalności będzie prawie na pewno bardzo trudnym zadaniem. Można by spróbować wybrać programy p i udowodnić VA(p) w bardziej sprytny sposób niż przez samo wyliczenie, aby poprawić wydajność bez niszczenia uniwersalności. Można by włączyć wszelkiego rodzaju pomysły, takie jak algorytmy genetyczne, zaawansowane dowodniki twierdzeń i wiele innych. Ale teraz mamy problem.

Obliczalność. Wydaje się, że przenieśliśmy problem AI na inny poziom. Ta zmiana ma pewne zalety (ale także pewne wady), ale nie przedstawia praktycznego rozwiązania. Niemniej jednak chcemy podkreślić, że sprowadziliśmy problem AI do (zwykłych) pytań obliczeniowych. Nawet najbardziej ogólne inne systemy, o których autor wie, zależą od pewnych (bardziej niż złożoność) założeń dotyczących środowiska lub nie jest jasne, czy są one rzeczywiście uniwersalnie optymalne. Chociaż pytania obliczeniowe są same w sobie wysoce skomplikowane, ta redukcja jest nietrywialnym wynikiem. Formalna teoria czegoś, nawet jeśli nie jest obliczalna, jest często wielkim krokiem w kierunku rozwiązania problemu i ma również własne zalety, a AI nie powinna się pod tym względem różnić (patrz poprzedni punkt). Elegancja. Wielu badaczy AI uważa, że inteligencja jest czymś skomplikowanym i nie można jej skondensować do kilku wzorów. Jest to raczej połączenie wystarczającej liczby metod i dużej ilości jawnej wiedzy we właściwy sposób. Z teoretycznego punktu widzenia się nie zgadzamy, ponieważ model AIXI jest prosty i wydaje się spełniać wszystkie potrzeby. Z praktycznego punktu widzenia zgadzamy się w następującym zakresie: Aby zmniejszyć obciążenie obliczeniowe, należy od samego początku zapewnić algorytmy specjalnego przeznaczenia (metody), prawdopodobnie wiele z nich związanych ze zmniejszeniem złożoności przestrzeni wejściowych i wyjściowych X i Y za pomocą odpowiednich metod przetwarzania wstępnego/końcowego.

Dodatkowa wiedza. Nie ma potrzeby włączania dodatkowej wiedzy od samego początku. Może być ona przedstawiona w pierwszych kilku cyklach w dowolnym formacie. Dopóki algorytm interpretujący dane ma rozmiar O(1), agent AIXI „zrozumie" dane po kilku cyklach. Jeśli środowisko μ jest skomplikowane, ale dodatkowa wiedza z sprawia, że $K(\mu|z)$ jest małe, można wykazać, że ograniczenie (17) redukuje się mniej więcej do $\ln 2 \cdot K(\mu|z)$, gdy $x_1 \equiv z$, tj. gdy z jest prezentowane w pierwszym cyklu. Algorytmy specjalnego przeznaczenia można przedstawić również w x_1, ale byłoby oszustwem stwierdzenie, że w AIXI nie zaimplementowano żadnych algorytmów specjalnego przeznaczenia. Granica między wdrożeniem a szkoleniem jest nieostra w modelu AIXI.

Szkolenie. Nie powiedzieliśmy zbyt wiele o samym procesie szkolenia, ponieważ nie jest on specyficzny dla modelu AIXI i był omawiany w literaturze w różnych formach i dyscyplinach . Przez proces szkolenia rozumiemy sekwencję prostych do złożonych zadań do rozwiązania, przy czym prostsze pomagają w nauce bardziej złożonych. Poważna dyskusja byłaby nie na miejscu. Powtarzając truizm, ważne jest oczywiście przedstawienie wystarczającej wiedzy ok i ocena wyjścia agenta y_k z r_k w rozsądny sposób. Aby zmaksymalizować zawartość informacyjną w nagrodzie, należy zacząć od prostych zadań i przyznać pozytywną nagrodę mniej więcej za lepszą połowę wyjść y_k.

Wielkie pytania

Ta podsekcja poświęcona jest wielkim pytaniom dotyczącym AI w ogóle, a w szczególności modelowi AIXI, z osobistym akcentem. O fizyce nieobliczalnej i mózgach. Istnieją dwa możliwe zarzuty wobec AI w ogóle, a zatem wobec AIXI w szczególności. Fizyka nieobliczalna (co nie jest zbyt dziwne) mogłaby uniemożliwić obliczeniową AI Turinga. Ponieważ przynajmniej świat, który jest istotny dla ludzi, wydaje się być głównie obliczalny, nie uważamy, że konieczne jest integrowanie urządzeń nieobliczalnych w systemie AI. (Sprytny i niemal przekonujący) argument Gödla Penrose'a, udoskonalający Lucasa, że fizyka nieobliczalna musi istnieć i jest istotna dla mózgu, ma (naszym zdaniem przekonujące) luki.

Ewolucja i liczba mądrości. Poważniejszym problemem jest ewolucyjny proces gromadzenia informacji. Wykazano, że „liczba mądrości" Ω zawiera bardzo zwartą tabelę 2^n nierozstrzygalnych problemów w pierwszych n cyfrach binarnych. Ω jest wyliczalna tylko wtedy, gdy czas obliczeń wzrasta szybciej z n niż jakakolwiek funkcja rekurencyjna. Ogromna moc obliczeniowa ewolucji mogła rozwinąć i zakodować coś takiego jak Ω w naszych genach, co znacząco kieruje ludzkim rozumowaniem. Krótko mówiąc: inteligencja może być czymś skomplikowanym, a ewolucja w jej kierunku od nawet sprytnie zaprojektowanego algorytmu o rozmiarze O(1) mogłaby być zbyt powolna. Ponieważ ewolucja już miała miejsce, moglibyśmy dodać informacje z naszych genów lub struktury mózgu do dowolnego/naszego systemu AI, ale oznacza to, że nadal brakuje ważnej części i że zasadniczo niemożliwe jest wyprowadzenie wydajnego algorytmu z prostej formalnej definicji AI.

Świadomość. W przypadku prawdopodobnie największego pytania, jakim jest świadomość, chcemy podać analogię fizyczną. Teoria kwantowa (pola) jest najdokładniejszą i najbardziej uniwersalną teorią fizyczną, jaką kiedykolwiek wymyślono. Choć rozwinięte już w latach 30. XX wieku, wielkie pytanie dotyczące interpretacji kolapsu funkcji falowej pozostaje otwarte. Choć jest to niezwykle interesujące z filozoficznego punktu widzenia, jest zupełnie nieistotne z praktycznego punktu widzenia. Wierzymy, że to samo dotyczy świadomości w dziedzinie sztucznej inteligencji: filozoficznie wysoce interesujące, ale praktycznie nieistotne. Czy świadomość zostanie kiedyś wyjaśniona, to już inna kwestia.

Wnioski

Głównym tematem rozdziału było opracowanie matematycznych podstaw sztucznej inteligencji. Nie jest to łatwe zadanie, ponieważ inteligencja ma wiele (często słabo zdefiniowanych) twarzy. Dokładniej rzecz biorąc, naszym celem było opracowanie teorii dla racjonalnych agentów działających optymalnie w każdym środowisku. W ten sposób poruszyliśmy różne obszary naukowe, w tym uczenie wzmacniające, algorytmiczną teorię informacji, złożoność Kołmogorowa, teorię złożoności obliczeniowej, teorię informacji i statystykę, indukcję Solomonowa, przeszukiwanie Levina, sekwencyjną teorię decyzji, teorię sterowania adaptacyjnego i wiele innych. Rozpoczęliśmy od spostrzeżenia, że wszystkie zadania, których rozwiązanie wymaga inteligencji, można naturalnie sformułować jako maksymalizację pewnej oczekiwanej użyteczności w ramach agentów. Przedstawiliśmy funkcjonalną (3) i iteracyjną (11) formułę takiego agenta teoretyczno-decyzyjnego, która jest wystarczająco ogólna, aby objąć wszystkie klasy problemów AI, co zostało wykazane na kilku przykładach. Głównym pozostałym problemem jest nieznany rozkład prawdopodobieństwa a priori μ środowiska(ów). Konwencjonalne algorytmy uczenia się są nieodpowiednie, ponieważ nie potrafią obsłużyć dużych (niestrukturyzowanych) przestrzeni stanów, nie zbiegają się w teoretycznie minimalnej liczbie cykli ani nie potrafią odpowiednio obsłużyć środowisk niestacjonarnych. Z drugiej strony uniwersalny rozkład a priori ξ (16) Solomonoffa, zakorzeniony w algorytmicznej teorii informacji,

rozwiązuje problem nieznanego rozkładu a priori dla problemów indukcji, jak wykazano w sekcji 3. Nie jest konieczna żadna jawna procedura uczenia się, ponieważ ξ automatycznie zbiega się do μ. Zjednoczyliśmy teorię uniwersalnej predykcji sekwencji z agentem teoretyczno-decyzyjnym, zastępując nieznany prawdziwy rozkład a priori μ odpowiednio uogólnioną uniwersalną półmiarą ξ w sekcji 4. Podaliśmy różne argumenty, że wynikowy model AIXI jest najbardziej inteligentnym, wolnym od parametrów i niezależnym od środowiska/aplikacji modelem, jaki jest możliwy. Zdefiniowaliśmy relację porządku inteligencji (definicja 5), aby nadać temu twierdzeniu ścisłe znaczenie. Ponadto omówiono możliwe rozwiązania problemu horyzontu. Opisaliśmy, jak model AIXI rozwiązuje różne klasy problemów. Obejmowały one przewidywanie sekwencji, gry strategiczne, minimalizację funkcji i, w szczególności, uczenie się uczenia nadzorowanego. Listę tę można by łatwo rozszerzyć na inne klasy problemów, takie jak klasyfikacja, inwersja funkcji i wiele innych. Główną wadą modelu AIXI jest to, że jest on nieobliczalny, a dokładniej, obliczalny jedynie asymptotycznie, co uniemożliwia implementację. Aby przezwyciężyć ten problem, skonstruowaliśmy zmodyfikowany model AIXItl, który jest nadal skutecznie bardziej inteligentny niż jakikolwiek inny algorytm ograniczony czasem t i długością l . Czas obliczeniowy AIXItl jest rzędu $t \cdot 2^l$. Sposób przezwyciężenia dużej stałej mnożnikowej 2^l został przedstawiony kosztem (niestety jeszcze większej) stałej addytywnej. Omówiono możliwe dalsze badania. Głównymi kierunkami mogłyby być udowodnienie ogólnych i szczególnych ograniczeń nagrody, użycie AIXI jako supermodelu i zbadanie jego relacji z innymi wyspecjalizowanymi modelami, a na koniec poprawa wydajności z rezygnacją z uniwersalności lub bez niej. Podsumowując, wyniki pokazują, że sztuczną inteligencję można ująć w eleganckiej teorii matematycznej. Poczyniono również pewne postępy w kierunku eleganckiej obliczeniowej teorii inteligencji.