

**Ks. Józef Kloch**

*Instytut Teologiczny, Tarnów*

## **OGRANICZENIA SZTUCZNEJ INTELIGENCJI W UJĘCIU HUBERTA L. DREYFUSA (CZ. 2)\***

Hubert L. Dreyfus w badaniach nad sztuczną inteligencją wydzielił cztery pola: formalizowanie rozgrywania gier, rozwiązywanie problemów, tłumaczenia językowe oraz rozpoznawanie obrazów. Niniejszy artykuł jest kontynuacją publikowanych już analiz<sup>1</sup> – dotyczy dwóch ostatnich dziedzin sztucznej inteligencji.

### **I. TŁUMACZENIA JĘZYKOWE – TRZECIE POLE SZTUCZNEJ INTELIGENCJI**

Wysiłki uczonych skupione nad formalizowaniem tłumaczeń językowych łączą się z filozoficznymi zagadnieniami związanymi z tworzeniem pojęć, uczeniem, syntaktyką i semantyką. Wittgenstein podkreślał znaczenie aktywności w tworzeniu pojęć przez człowieka; proces ten musi angażować podmiot i nie ma w nim miejsca na bierność. Pokazanie dziecku stołu i powiedzenie „brązowy” nie doprowadzi go do wniosku, że chodzi jedynie o kolor rzeczy a nie samą rzecz. Dopiero aktywne zaangażowanie dziecka i uczącego w proces uczenia doprowadzi do tworzenia pojęć w jego umyśle. Poglądy Wittgensteina sprawdziły się w odniesieniu do komputerowych programów. Nauczanie i uczenie się musi być aktywne, inaczej jest tylko wzbudzaniem reakcji na bodziec i przypomina odruchy warunkowe analizowane przez Pawłowa.

Próby ułożenia programu tłumaczącego język naturalny ujawniły wielką złożoność syntaktyki i semantyki, jakiej wcześniej nie podejrzewano. Mechaniczny słownik, zasady gramatyki i kontekst wynikający ze zdania nie wystarczyły do sformalizowania tłumaczeń. Decydujące znaczenie w poprawności tłumaczenia, zwłaszcza tekstów naukowych, ma kontekst globalny i odwołanie się do obrzeży świadomości człowieka. Dla Dreyfusa problemy związane z formalizowaniem

---

\* Artykuł stanowi drugą część pracy licencjackiej, napisanej pod kierunkiem bpa prof. dra hab. J. Życińskiego na Wydziale Filozoficznym PAT w Krakowie.

<sup>1</sup> Por. J. Kloch, *Ograniczenia sztucznej inteligencji w ujęciu Huberta L. Dreyfusa (cz. 1)*, TarnStT 13 (1994) s. 51-81

tłumaczenia języka naturalnego były kolejnym przykładem specyficznego przetwarzania informacji, które zachodzi w ludzkim umyśle.

Jest rzeczą interesującą, że gdy uczeni zdali sobie sprawę ze stopnia złożoności tej dziedziny AI (*artificial intelligence*), nie usiłowali oni podtrzymać optymizmu przez pokładanie nadziei w przyszłości mającej rozwiązać wszelkie problemy, lecz uznali niemożność rozwiązań ogólnych na tym etapie badań. Analizy związane z tłumaczeniem języka naturalnego przyjęły następnie postać tzw. *cognitive linguistics*. Jedną z bardziej interesujących teorii tego nurtu jest *Conceptual Dependency* R.C. Schanka.

### 1. Formalizacja tłumaczeń językowych

W dziedzinie *language translating* rozwiązany został do końca problem mechanicznego słownika; twórcą tego programu był Anton Oettinger. Oprócz niego zabierali głos m.in. Edward Feigenbaum, Earl Hunt i Carl Hovland oraz Robert Lindasay. Ich badania nie doprowadziły jednak do uporania się z głównym problemem – sformalizowaniem tłumaczenia języka naturalnego. Wysiłki uczonych doprowadzały do nowych pytań a zakres analiz był coraz bardziej zawężany. W trakcie prac naukowcy sygnalizowali kolejne problemy, które należało rozwiązać w przyszłych programach komputerowych.

W pierwszym, asocjacyjnym podpolu *language translating* Dreyfus umieścił w swojej klasyfikacji<sup>2</sup> tłumaczenie słowa po słowie, czego przykładem jest mechaniczny słownik. Wyświetla on znaczenie określonego słowa a metodą działania takiego słownika jest przeszukiwanie zbioru słów według wzorca. Na wyjściu komputera podawane są wszystkie znaczenia wprowadzone do słownika, niezależnie od związków syntaktycznych czy semantycznych.

Do czołowych badaczy zajmujących się tłumaczeniem języka należy A.G. Oettinger. Jedną z jego głównych prac był mechaniczny słownik (1954). Sporządzenie go było możliwe nawet wówczas, gdy prace nad *hardware* oraz *software* nie były jeszcze zbyt zaawansowane. Częściom słów, całym słowom lub grupom słów przypisywane były ich znaczenia w innym języku; w ten sposób dane słówko lub zwrot mógł być tłumaczony np. z rosyjskiego na angielski i odwrotnie. Sukces mechanicznego słownika Oettingera ożywił nadzieje na rychłe sporządzenie programu automatycznie tłumaczącego teksty z jednego języka na drugi. Jakość tych przekładów miała być bardzo wysoka, a pierwsze próby obejmowały nie bez kozery język angielski i rosyjski. Przydatność wojskowa i naukowa takiego programu była oczywista; należało tak zaprogramować komputer, by otrzymać idiomatyczną anglojęzyczną wersję tekstu rosyjskiego.

Prace badawcze nad tłumaczeniem języka naturalnego trwały dziesięć lat. Zajmowało się tym zagadnieniem 5 rządowych agencji, które miały do dyspozycji

---

H.L. Dreyfus, *What Computers Can't Do. The Limits of Artificial Intelligence*, New York: Harper Colophon Books, <sup>2</sup>1979, s. 292-293. W dalszej części artykułu w odniesieniu do tej książki stosuję skrót WCCD.

20 mln dolarów. W pierwszym roku badań uczeni odnieśli dużo sukcesów i wielu badaczom wydawało się, że w zasięgu ręki leży rozwiązanie wszystkich problemów związanych z automatycznym tłumaczeniem. Z perspektywy czasu można zrozumieć ówczesny optymizm, który okazał się jednak iluzją. Optymistyczne nastawienie powstało wskutek stosunkowo szybkiego rozwiązania wielu zagadnień związanych z programowaniem; jednak daleko było jeszcze do otrzymania właściwego tłumaczenia charakteryzującego się wysoką jakością. Uczeni poradzili sobie rzeczywiście z wieloma problemami, ale należały one do najprostszych. Dlatego tak wiele i tak szybko rozwiązali. Pozostało jednak kilka zagadnień i te okazały się najtrudniejsze. Analizowanie ich ukazało głębię złożoności syntaktyki i semantyki. W ocenie stanu badań nad automatycznym tłumaczeniem języka Oettinger pisał: „Pozostaje nie rozwiązany główny problem dobierania odpowiedniego znaczenia dla słowa źródłowego na podstawie kontekstu, (...) który czytający człowiek określa jednoznacznie”<sup>3</sup>

Dopiero w połowie lat sześćdziesiątych zdano sobie sprawę z nierozwiązywalności tego zadania w odniesieniu do programu komputerowego. Kropkę nad „i” postawił w 1966 r. raport Narodowej Akademii Nauk, który powstał po porównaniu tłumaczenia zrobionego przez maszynę oraz człowieka. Raport stwierdzał, że do tłumaczenia tekstów naukowych w ogóle nie można stosować mechanicznego tłumaczenia oraz że aktualne wyniki badań mówią o braku perspektyw dla maszynowego tłumaczenia, a przyszłe rozwiązania trudno przewidzieć.<sup>4</sup>

Odnosnie do dziedziny *language translating* Dreyfus nie widzi możliwości formalizowania programów, które można by zaliczyć do podpoła II (Simplex-Formal) oraz III (Complex-Formal). Do IV, nieformalizowalnego pola, zalicza tłumaczenie języka naturalnego. Badania przełomu lat pięćdziesiątych oraz sześćdziesiątych potwierdzają jego opinię. Zawiodła go natomiast intuicja co do dwóch pozostałych pól.

Po 10 latach wczesny optymizm co do tłumaczenia maszynowego ustąpił miejsca bardziej wyważonym poglądom. Uczeni rozpoczęli pisywanie programów operujących ograniczoną liczbą rodzajów zdań oraz bardzo ograniczonym słownictwem; również tematyka zagadnień została bardzo zawężona. Przekład jednego języka naturalnego na inny przestał być już dążeniem uczonych; obecnie starali się oni, by komputer rozumiał zwracanie się do niego w języku naturalnym oraz by dawał on odpowiedzi na konkretne pytania. Aby tak się stało, komputer musiał akceptować i interpretować poprawnie słowa danego języka naturalnego. W badaniach nad tą dziedziną AI został wykonany typowy, lecz tym razem uzasadniony zwrot ku próbom uczenia maszyn. Jeśli bowiem komputer ma odpowiadać na pytania stawiane w języku naturalnym, musi posiadać umiejętność uczenia się o otaczającym go świecie. Pierwszym przybliżeniem tej sztuki był w opinii E. Feigenbauma

<sup>3</sup> A. Oettinger, *The State of the Art of Automatic Language Translation: An Appraisal*, w: *Beiträge zur Sprachkunde und Informationsverarbeitung*, herausg. H. Marchl, Bd. I, Heft 2, München: Oldenbourg Verlag 1963, s. 21.

<sup>4</sup> *Language and Machines*, National Academy of Sciences, Washington D.C. 1966, s. 29.

program<sup>5</sup> symulujący werbalne uczenie się nonsensownych trzyliterowych wyrazów. *Elementary Perceiver and Memorizer* (EPAM, 1961) należał do klasy modeli przetwarzania informacji, które były modelami procesów myślowych ale nie neurologicznej struktury mózgu. Według tej koncepcji organy zmysłów były kanałami wejścia, a urządzeniami wyjściowymi – organy efektorowe. Mechanizm przetwarzania danych był pojmowany jako szeregowy. Zmienne o charakterze przypadkowym nie odgrywały w modelach tego typu żadnej roli – z zasady były one deterministyczne. Eksperymenty przeprowadzane z EPAM-em nosiły nazwę prezentacji sylab skojarzonych parami oraz kolejnej prezentacji przewidywanych sylab (n+1, n+2 etc.). Koncepcyjnie EPAM można podzielić na dwa podprogramy: uczenia się i wykonawczy. W etapie uczenia się EPAM kojarzył i rozróżniał elementy, a w etapie wykonawczym EPAM generował reakcję na bodziec. Program najpierw analizował bodziec (trzyliterowe słowa) – szukał istotnych cech bodźca, aby znaleźć jego obraz. Następnie szukał wskazówek związanych z obrazem i odnajdywał reakcję na bodziec, po czym generował reakcję. Eksperymenty dotyczyły:

(...) różnorodnych „jakości zmysłowych” dla modelu EPAM, odpowiadających „wzrokowemu” wejściu i „pisemnemu” wyjściu, „słuchowemu” wejściu i „ustnemu” wyjściu (...). Tak więc EPAM może, tak jak każdy z nas, słyszeć swym „wewnętrznym uchem” to, co „widzi” swymi „oczyma wyobraźni”<sup>6</sup>

Feigenbaum wspomina też o kilku eksperymentach związanych z asocjacyjnym uczeniem EPAM-u czytania. Píše też o tym, że EPAM nie był w stanie nauczyć się wyrazów, w których dana sylaba występowała dwukrotnie.

Badania nad programem EPAM nie stanowią według Dreyfusa próby uczenia maszyny o świecie rzeczywistym, co byłoby drogą do rozumienia i przetwarzania języka naturalnego. Dla autora *What Computers Can't Do* są one raczej symulowaniem odruchów warunkowych Pawłowa, czyli kontynuacją prac Oettingera z roku 1952,<sup>7</sup> choć w nowej szacie programu.

Analizy Hunta i Hovlanda<sup>8</sup> związane z symulowaniem wytwarzania pojęć u człowieka były kolejnymi badaniami nad uczeniem komputerów o otaczającym świecie; z tymi pracami (1961) wiązano duże nadzieje. Program napisany w języku IPL-V potrafił rozpoznawać pojęcia typu: „mały-biały-kwadrat”, jednak autorzy sami orzekli, że nie można potwierdzić ani odrzucić przydatności tego typu symulacji w psychologii.

<sup>5</sup> E. Feigenbaum, *Symulacja uczenia się werbalnego*, w: *Maszyny matematyczne i myślenie*, pod red. E.A. Feigenbauma i J. Feldmana, Warszawa: PWN 1972, s. 292-306. W dalszej części artykułu w odniesieniu do wymienionej pracy zbiorowej stosuję skrót MMiM.

<sup>6</sup> Tamże s. 305-306.

<sup>7</sup> Jego pierwsze prace dotyczyły programu naśladowającego klasyczne odruchy warunkowe Pawłowa; program ten został zrealizowany na komputerze EDSAC w 1952 r. i pozwalał na modelowanie prostych reakcji.

<sup>8</sup> Por. E.B. Hunt, C.I. Hovland, *Programowanie modelu wytwarzania pojęć u człowieka*, w: MMiM, s. 306-321.

Badania R. Lindsaya,<sup>9</sup> dotyczące programu rozbioru zdania oraz programu analizy semantycznej, jeszcze raz zwróciły uwagę na problem znaczenia wyrazów w zależności od konotacji i implikacji. Zilustrował to zagadnienie programem komputerowym gromadzącym syntaktyczne reakcje i wydobywającym oraz gromadzącym semantyczne implikacje. Zaznaczył przy tym, że jego program dysponował jedynie ograniczonymi możliwościami i miał tylko posłużyć zasygnalizowaniu problemów, które trzeba rozwiązać w przyszłych programach.

Inne nieco zagadnienie dotyczące uczenia maszyn poruszyli uczeni, budując komputer nazwany Minerwa z Kent.<sup>10</sup> Urządzenie to było zdolne przyswoić sobie tyle wiedzy, ile przeciętny człowiek w ciągu 70 lat życia. Do takiego podejścia można odnieść zastrzeżenie Dreyfusa i Wittgensteina – czy mechaniczne gromadzenie informacji można nazwać uczeniem? Czy bez aktywnego udziału podmiotu proces ten może w istocie zmierzać do uczenia maszyn?

## 2. Uwagi Dreyfusa na temat ograniczeń związanych z tłumaczeniami językowymi

Prace nad tłumaczeniami języka ujawniły raz jeszcze rolę pobocznej świadomości w ludzkim procesie przetwarzania informacji. Świadomość ta ma decydujące znaczenie w określaniu kontekstu znaczenia poszczególnych słów i zdań. W zagadnieniu *language translating* powraca też problem uczenia się; proces ten ma wielkie znaczenie w rozumieniu języka naturalnego.

Znaczenie danego wyrażenia czy słowa nie jest jednoznaczne, zmienia się w zależności od kontekstu. Niemniej w określonej sytuacji znaczenie jest jednoznaczne i dzięki temu określone zdanie czy wyraz jest właściwie rozumiane przez ludzi. Wittgenstein uważał,<sup>11</sup> że dla pojęć nie istnieje żadna rzeczywista definicja. Gdyby przyjąć przeciwne założenie, należałoby też sądzić, że gdy dzieci bawią się piłką, to robią to zawsze według ścisłych reguł. Podobne do Wittgensteina poglądy mieli niektórzy uczeni zajmujący się rozumieniem komputerów.<sup>12</sup> Przyjmują oni, że naturalny język ma charakter pozbawiony ścisłych reguł; ma on nieograniczoną liczbę kombinacji, a ich znaczenia są wnioskowane ze skończonego zbioru zasad rządzących rozumieniem zdań czy poszczególnych wyrazów.

Wydawałoby się, że do tłumaczenia języka przez komputer potrzeba jedynie odpowiedniego słownika mechanicznego i reguł gramatycznych, a pisemny kontekst dostarczy innych informacji istotnych w tłumaczeniu. Jednakże nawet najbardziej kompletny zbiór słów i ich znaczeń oraz najbardziej skomplikowane reguły gramatyczne, a także pisemny kontekst są zbyt ubogimi informacjami, by tłumaczenie stało na odpowiednio wysokim poziomie.

---

<sup>9</sup> R. L i n d s a y, *Pamięć dedukcyjna – jako podstawa działania maszyn rozumiejących język naturalny*, w: MMiM, s. 220-235.

<sup>10</sup> Por. M. H o l y Ń s k i, *Sztuczna inteligencja*, Warszawa: Wiedza Powszechna 1979, s. 147-148.

<sup>11</sup> L. W i t t g e n s t e i n, *The Blue and Brown Books*, Oxford: Basil Blackwell 1960, s. 25.

<sup>12</sup> Por. M. K o c h e n [i in.], *Computers and Comprehension*, RAND Corporation, RM-4065-PR, April (1964) s. 12.

Omówiony dokładnie przez Dreyfusa<sup>13</sup> argument Bar-Hillela mówi o nieskończonej liczbie istotnych wskazówek odnośnie do rozumienia zdania czy słowa; w określonym kontekście nie występuje większa część z możliwych do pomyslenia wieloznaczności. Kontekst sterujący percepcją pozwala na poprawne słyszenie zdania; ponieważ zaś zdania nie są odbierane bez kontekstu, powoduje to zawężenie zakresu znaczeń zdania. O ile strumień słów w określonym kontekście jest właściwie interpretowany przez ludzi, o tyle dla komputerów strumień ten jest nie lada problemem.

Dreyfus wylicza elementy<sup>14</sup> ograniczające wieloznaczność wyrażeń; są nimi cele pozajęzykowe oraz zaangażowanie ludzi w konkretnej sytuacji. Cele pozajęzykowe, które nie muszą być precyzyjnie określone czy nawet możliwe do określenia, dostarczają pewnych wskazówek redukujących wieloznaczność wyrażeń do skali wymaganej w danym zdaniu. Zaangażowanie człowieka w określoną sytuację stanowi kolejną wskazówkę – znaczenie wyrażenia staje się jednoznaczne w szczegółowej sytuacji.

Dreyfus uważa, że aby wyeliminować wieloznaczności człowiek ma zdolność do stosowania kontekstu globalnego; poboczna świadomość korzysta tu ze wskazówek kontekstu w odniesieniu do wielu prawdopodobnych interpretacji. W przypadku komputera wszystkie te interpretacje musiałyby znaleźć się na wyjściu, podczas gdy ludzkie poczucie kontekstu sytuacji pozwala na eliminowanie większości możliwych interpretacji bez konieczności brania ich pod uwagę. Autor *What Computers Can't Do* nazywa tę zdolność do zawężania spektrum możliwych znaczeń przez ignorowanie tego, co byłoby wieloznaczne bez kontekstu, tolerancją wieloznaczności (*ambiguity tolerance*). Zdolność taka jest właściwa ludzkiej naturze i nie przysługuje komputerowi, co stanowi, jak podkreśla Dreyfus, kolejną barierę dla sztucznej inteligencji.

W zagadnieniu *language translating* istotną rolę odgrywa problem uczenia się; w przypadku tej dziedziny AI, odwołanie się do uczenia nie jest próbą ominięcia trudności napotykanych na coraz to bardziej zaawansowanym etapie badań. Człowiek uczy innego człowieka sztuki tłumaczenia, dając mu od czasu do czasu właściwą wskazówkę. Istnieją co prawda pewne reguły, ale nie są one stosowane sztywno i tylko doświadczeni tłumacze umieją z nich korzystać. O ile człowiek potrafi nauczyć się poprawności w oddawaniu myśli autora, o tyle systemy komputerowe napotykały w tym przypadku na trudności.

Program *Elementary Perceiver and Memorizer* symulował uczenie się wzajemnego odniesienia bezsensownych sylab, co Feigenbaum<sup>15</sup> nazywał uproszczonym przypadkiem werbalnego uczenia się. Dreyfus nie zgadza się jednak z taką oceną – tego rodzaju uczenie nie jest dla niego przypadkiem nauki języka.<sup>16</sup> Próby takie porównuje on do osiągnięcia czegoś podobnego do odruchu warunkowego

<sup>13</sup> WCCD, s. 214-218.

<sup>14</sup> WCCD, s. 108-109.

<sup>15</sup> Za: WCCD, s. 109.

<sup>16</sup> WCCD, s. 109nn.

Pawłowa. Wielokrotne powtarzanie dwóch bezsensownych wyrazów, a następnie zapalenie odpowiednio czerwonego światła dla pierwszego i zielonego światła dla drugiego wyrazu, mogłoby doprowadzić do rozróżniania przez komputer par wyraz-swiatło. Ale taki eksperyment zakłada całkowitą bierność podmiotu – on nie uczy się niczego nowego, tylko jest w nim wyzwalana reakcja na bodziec.

W przypadku programu GPS porównanie protokołu człowieka ze sprawozdaniem maszyny ujawniło niezgodności w kilku miejscach. Przeprowadzenie podobnego eksperymentu z EPAM nie ujawniło żadnych niezgodności. Ale Dreyfus uważa ten sukces za wątpliwy – ów pozytywny przypadek naśladowania procesu uczenia się symuluje proces nie będący rzeczywistym uczeniem się, bo nie wymagający rozumienia.

Człowiek wypowiada zdania i rozumie je w danych sytuacjach; aby komputer potrafił rozumieć bieżące wypowiedzi oraz tłumaczyć język naturalny, musiałby uczyć się o świecie. Bar-Hillel<sup>17</sup> stawia warunek – programy komputerowe muszą umożliwiać komputerowi uczenie się, by był on zdolny do dokonywania tłumaczeń na wysokim poziomie. Opinia ta nie przekreśla co prawda możliwości uczenia komputerów, ale wyraźnie przedstawia ograniczenia, którym podlegają te maszyny.

Zaangażowanie w naukę nie polega na tylko automatycznym powtarzaniu bezsensownych wyrazów – jest ono bardziej złożone niż warunkowe odruchy Pawłowa. Ucząc dziecko np. nowego wyrazu nie wystarczy wskazać na daną rzecz i wypowiedzieć ów wyraz; nie wiadomo bowiem, czy jest to nazwa własna rzeczy czy jedna z cech rzeczy. Zaangażowanie pozwala dziecku ograniczyć możliwe znaczenia wyrazu, lecz dla komputera jest to niemożliwe. W wypadku EPAM-u wyzwalano tylko reakcję komputera na bodziec i trudno nazwać tę reakcję wynikiem uczenia się komputera. Coś podobnego robiono z podmiotem, ale on sam niczego się nie uczył i nie był zaangażowany w proces uczenia. Niemożność symulowania przy pomocy programu procesu aktywnego uczenia się to – według Dreyfusa – kolejne ograniczenie sztucznej inteligencji. Dane są wprowadzane do komputera na innej zasadzie niż uczy się dzieci nowych pojęć.

Podczas uczenia język nie jest i nie może być precyzyjnie zdefiniowany (argument Bar-Hillela). Uczenie się, w przeciwieństwie do zapamiętywania i powtarzania, wymaga kontekstu; to on właśnie wprowadza jednoznaczność w rozumieniu języka naturalnego. Kontekst globalny, wymagany do poprawnego tłumaczenia języka naturalnego, leży poza zasięgiem komputerów.

### 3. Podsumowanie

Początkowy optymizm, spowodowany sukcesem związanym z mechanicznym słownikiem A. Oettingera, rokował szybkie uporanie się z ogólnym programem tłumaczącym jeden język naturalny na inny. Pierwszy etap prac nad tym programem

---

<sup>17</sup> Y Bar-Hillel, *The Present Status of Automatic Translation of Languages*, w: *Advances in Computers*, ed. F.L. Alta, New York: Academic Press 1966, t. 1, s. 94.

podtrzymał optymistyczne nastawienie uczonych, lecz nie na długo. Napotkanie problemów związanych ze złożonością syntaktyki i semantyki doprowadziło w końcu do zaniechania badań. Uczeni wykazali w tym wypadku dużo rozsądku i nie dopatrywali się w przyszłości rozwiązań wszelkich problemów. Choć zawężili zakres badań, to jednak ich prace częściej rodziły nowe pytania niż przynosiły oczekiwane odpowiedzi.

Według Dreyfusa, świadomość kontekstu oraz tolerancja wieloznaczności stanowią dla komputerów ograniczenia nie do pokonania. To one okazały się powodem zahamowania badań nad tłumaczeniem języka naturalnego, choć wcześniej miał miejsce sukces mechanicznego słownika. Zdolność uczenia się języka zakłada złożone formy ludzkiego przetwarzania informacji potrzebne do zrozumienia języka;<sup>18</sup> aktywny proces uczenia się nie jest możliwy do symulowania na komputerze i to jest jeszcze jedna bariera, na którą napotykają maszyny.

Wnioski Dreyfusa zgadzają się z opiniami Bar-Hillela i Oettingera. Autor mechanicznego słownika, badając systemy wytwarzające wszystkie interpretacje zdania zgodne z gramatyką, stwierdził:

Działanie obecnych analizatorów ujawniło dużo wyższy stopień wieloznaczności składniowej w językach angielskim i rosyjskim niż zakładano. To i związany z nią brak wyraźnej granicy między wyrażeniami gramatycznymi a niegramatycznymi, stwarza poważne pytania odnośnie do możliwości efektywnej, w pełni automatycznej manipulacji językiem angielskim i rosyjskim dla celów tłumaczeń lub odzysku informacji.<sup>19</sup>

Od Oettingera, twórcy mechanicznego słownika, należałoby raczej oczekiwać tłumaczenia się przejściowymi trudnościami, które w przyszłości będą rozwiązane; tak postępowali choćby Newell, Shaw i Simon ilekroć napotykali problemy. Natomiast autor programu EPAM podkreśla złożoność semantycznego procesu pozwalającego ludziom na jednoznaczne interpretowanie zdań oraz zaznacza, że dotąd proces ten nie jest dostatecznie zgłębniony.<sup>20</sup>

Jednoznaczność rozumienia przez ludzi języka naturalnego to dla Dreyfusa<sup>21</sup> jeszcze jeden przykład znaczenia pobocznej świadomości. Człowiek najprawdopodobniej nie jest świadomy, że przy określaniu znaczenia reaguje na wiele wskazówek docierających do niego; z drugiej strony nic nie wskazuje na to, że człowiek nieświadomie analizuje każdą z tych wskazówek. Jedna i druga ewentualność nie jest możliwa do symulowania przez komputerowy program, niezależnie od tego, czy byłby on równoległy czy sekwencyjny.

<sup>18</sup> Por. W Lubaszewski, *Rozumienie tekstu przez komputer. Wprowadzenie do problematyki i inteligentny program komputerowy*, Kraków: Ośrodek Badań Prasoznawczych 1990, s. 10.

<sup>19</sup> Oettinger, jw. s. 26.

<sup>20</sup> Tamże.

<sup>21</sup> WCCD, s. 108.



## II. ROZPOZNAWANIE OBRAZÓW – CZWARTE POLE SZTUCZNEJ INTELIGENCJI

Angielskie określenie *pattern recognition* jest trudne do oddania w języku polskim. Zwykle tłumaczy się je jako rozpoznawanie obrazów, co sugeruje przede wszystkim związek z artystycznymi przejawami aktywności człowieka. Trzeba jednak zaznaczyć, że czwarte pole AI dotyczyło początkowo identyfikacji cyfr i liter, znaków graficznych, rozumienia mowy, rozpoznawania sytuacji przestrzennej, a dopiero potem obrazów, ilustracji, rysunków etc. Choć nazwa tego nie podkreśla, w *pattern recognition* uczeni dążyli do wyposażenie komputerów w percepcyjne zdolności człowieka.

Rozpoznawanie obrazów jest rozważane przez Dreyfusa jako ostatnie, czwarte z kolei pole sztucznej inteligencji. Rozwiązanie wielu problemów w trzech pozostałych polach AI zakładało sukces w *pattern recognition*, lecz ono właśnie skupiło w sobie wszelkie trudności napotkane uprzednio.<sup>22</sup> Feigenbaum i Feldman również zaliczają rozpoznawanie obrazów do najtrudniejszych prac nad sztuczną inteligencją, i dostrzegają „bliskie pokrewieństwo między pracami z zakresu rozpoznawania obrazów oraz studiami świadomego zachowania się”<sup>23</sup>

W jaki sposób przebiega w umyśle człowieka rozpoznawanie danej rzeczy, np. stołu? Czy człowiek za każdym razem przeprowadza klasyfikację przedmiotu? Czy tworzy koncepcję stołu i za każdym razem sprawdza cechy, a jeśli tak, to wszystkie czy też niektóre, decydujące o tym, że dana rzecz jest tym, czym jest, a nie czymś innym? Co decyduje o podobieństwie jednej rzeczy do innej? Tego typu problemy analizowali m.in. A. Gurwitsch i L. Wittgenstein. Dreyfus stosuje ich rozróżnienia w *What Computers Can't Do* oraz odwołuje się do koncepcji pobocznej świadomości, roli zależności kontekstowej i intuicji. Te trzy składowe są istotne dla ludzkiego rozpoznawania obrazów i według Dreyfusa nieprzekładalne na system komputerowy.<sup>24</sup>

Każdy problem z zakresu *pattern recognition* wymaga od komputera postępowania niemal analogicznego do pracy nad rozpoznawaniem liter; obejmuje ono znormalizowanie obrazu, porównanie jego cech z zapamiętanymi wzorcami oraz ocenienie, do którego jest najbardziej podobny. Chodzi więc o przetworzenie danych wejściowych w nazwy lub opisy na wyjściu. Mniej ważny był rozpoznawany element, istotniejsza była metoda rozpoznawania; było nią szukanie uprzednio określonych cech topologicznych znaków, które miały być rozpoznane oraz kontrolowanie tych cech w porównaniu do obecnych lub wyuczonych definicji każdej litery. Jak znaleźć istotne cechy, tj. takie, które pozostają ogólnie niezmiennymi wariantami wymiarów i innych przekształceń? Rozwiązania przynoszące sukcesy miały miejsce tam, gdzie rozpoznawanie zależało od małej liczby specyficznych cech.

---

WCCD, s. 97

<sup>23</sup> E.A. Feigenbaum, J. Feldman, *Rozpoznawanie obrazów*, w: MMiM, s. 236.

<sup>24</sup> WCCD, s. 120-125.

Chcąc osiągnąć lepsze wyniki, badacze musieli rozwiązać tzw. problem prezentacji wiedzy. Należało rozstrzygnąć, które cechy są reprezentatywne dla danego obiektu, decydujące o określeniu, że dana rzecz jest tym a nie czymś innym. W jaki jednak sposób przekazać komputerowi ludzkie umiejętności rozpoznawania obrazów? Jak sformalizować rozpoznawanie, by było ono efektywne i wolne od pomyłek? Żaden z programów nie stanowił przełomu w *pattern recognition*, każdy był małym sukcesem inżynierskim, rozwiązaniem *ad hoc* specyficznego problemu bez ogólnego zastosowania.<sup>25</sup> Metody techniczne mają wbudowane ograniczenia; istnieje pewien poziom złożoności, powyżej którego inżynierski zestaw tricków nie wystarcza, by osiągnąć rezultaty. Ponadto często obrazy nie są obiektywnie jednoznaczne. Dotyczy to np. rysunków, których znaczenie może rozstrzygnąć jedynie autor rysunku (wazon czy dwa zwrócone do siebie profile?) lub grafik Eschera czy rysunków obiektów, których przestrzenna realizacja jest niemożliwa. W takich wypadkach jeszcze bardziej ujawnia się specyficzne ludzkie przetwarzanie informacji w rozpoznawaniu obrazów, będące poza zasięgiem formalizmów.

### 1. Badania nad rozpoznawaniem obrazów

Wstępem do rozpoznawania obrazów były badania nad sieciami neuronowymi, które rozpoczęły się w 1949 r.<sup>26</sup> Prace nad formalizowaniem *pattern recognition* zaczęły się stosunkowo późno pod koniec lat pięćdziesiątych. W czasach, gdy Dreyfus pisał *What Computers Can't Do*, istniały dwa główne rodzaje programów typu *pattern recognition*. Pierwszy zawierał szczegółowe instrukcje przetwarzania cech obrazów wejściowych; programy te nie były w stanie rozpoznawać obrazów skomplikowanych i różniących się jedynie szczegółami. Programiści wstrzymali się od wprowadzania do programów coraz to większego stopnia skomplikowania; byli bowiem świadomi, że każde takie przystosowanie zamiast stanowić lepszy stopień rozpoznawania, byłoby wprowadzaniem *ad hoc* skomplikowanych rozwiązań. Te modyfikacje miałyby znaczny wpływ na wydłużenie czasu potrzebnego do rozpoznawania obrazu

Drugi rodzaj programów był opracowywany jako *tabula rasa* i miał zdolność do zmian pewnych parametrów czyli do swego rodzaju „uczenia się” Wówczas programy te nie dysponowały jednak metodami gromadzenia doświadczenia. Zbieranie informacji, dotyczących macierzy wejściowej, było niewystarczającym rozwiązaniem, bowiem jakakolwiek zmiana w obrazach powodowała, że program był zawodny. Odmianą tego rodzaju programów był program z n-elementowym operatorem ze sprzężeniami przypadkowymi. Ale metoda ta miała taką samą skuteczność jak metoda probabilistyczna, wykorzystująca pojedyncze elementy,

<sup>25</sup> WCCD, s. 97.

<sup>26</sup> 1949 – W McCulloch: badania nad systemem nerwowym dżdżownicy. porównania z systemem komputera; 1958 – F Rosenblatt: perceptrony – modele nerwów wzrokowych; 1960 – L.G Roberts: modyfikacja perceptronu przez zastosowanie tzw. połączeń adaptatywnych usprawniających reakcje; 1972 – J. Race: nowa wersja perceptronu.

i zawodziła w badaniu obrazów znacznie różniących się od tych, na których maszyna „nabierała doświadczenia”

W 1959 r. rozpoczęto prace od najprostszych zadań; znaki alfabetu Morse’a są łatwe do identyfikacji, co skłoniło uczonych do ułożenia programu automatycznie dekodującego ów alfabet. Owocem wysiłków był MAUDE (*Morse Automatic Decoder*), opracowany przez grupę z MIT pod kierownictwem Bernarda Golda. Kropkom, kreskom i przerwom przyporządkowywane były odpowiednie liczby. Dzięki temu MAUDE wychwytywał prawidłowości rządzące liczbami i potrafił dekodować tekst nawet wtedy, gdy nie zostały w nim zachowane odpowiednie przerwy między wyrazami i literami. MAUDE nie osiągał jednak wyników, do jakich dochodzili radiotelegrafści z kilkuletnią praktyką – słyszą oni nie tyle kropki i kreski, lecz odbierają litery jako całość. MAUDE wykorzystywał informację zależną od kontekstu lecz w stopniu daleko mniejszym, niż wyszkolony operator. Zresztą chodziło tu o sposób porozumiewania się z maszyną – należało nauczyć ją dekodowania sygnałów dźwiękowych na litery i cyfry. Badania nad programem MAUDE wyznaczyły kierunek dalszych prac.

MAUDE stanowił przykład programu asocjacyjistycznego<sup>27</sup> z czwartego pola sztucznej inteligencji – rozpoznawania obrazów. Otrzymując dane na wejściu, program przetwarzał je według oderwanych od sytuacji, ściśle określonych wzorców, od których nie było żadnych odstępstw. Przetwarzanie odbywało się na zasadzie przeszukiwania listy wzorców; na wyjściu otrzymywany jest ciąg liter i cyfr.

Kolejnym krokiem w badaniach nad *pattern recognition* było rozpoznawanie pojedynczych obrazów ściśle określonych i nie zmieniających się, np. odczytywanie znaków napisanych przy pomocy maszyny do pisania. Odpowiedni algorytm poszukiwał cech świadczących o przynależności znaku lub jego fragmentu do danej kategorii. Ten typ programów określa Dreyfus jako sformalizowane prosto.<sup>28</sup> Przykładem podpola *Simple-Formal* jest *Pandemonium* Ulrika Neissera i Olivera Selfidge; prace nad nim prowadzone były od 1959 r.<sup>29</sup> Program sprawdzał cechy danego znaku i określał, co to za litera. Pierwsza wersja programu rozpoznawała cztery litery, druga – dziesięć. *Pandemonium* miał dwa różne poziomy operacji – pierwszy dokonywał przesunięcia obrazu, drugi porównywał obraz z opisem, do którego należał nie tylko kształt litery, ale również pewne charakterystyczne cechy typu poprzeczka, linia pionowa, wklęsłość od góry etc. Niektóre cechy nie mogły być jednak przedmiotem prostej dychotomii. Np. litera A zawiera poprzeczkę, litera O nie ma jej, ale co w przypadku litery B? *Pandemonium* radził sobie z tym zagadnieniem w ten sposób, że informacja nie miała charakteru binarnego lecz była reprezentowana przez ilościową przewagę różnych cech. Na obraz była nakładana siatka złożona z 1024 kwadratów. Program zawierał sześć etapów prze-

<sup>27</sup> Por. WCCD, s. 292-293.

<sup>28</sup> Tamże.

<sup>29</sup> Artykuł na temat *Pandemonium* został opublikowany w roku następnym: *Pattern recognition by machine*, Scientific American 203 (1960) August s. 60-68; polski przekład: *Rozpoznawanie obrazów przez maszynę*, w: MMiM, s. 237-250.

tworzania danych: wprowadzenie obrazu, wygładzenie go, przegląd cech, porównanie z wykazem cech, obliczenie prawdopodobieństw oraz decyzję. *Pandemonium* potrafił dokonywać drobnych korekt znaku, nie umiał jednak sobie poradzić ze zwartym tekstem i znaki trzeba było przedstawiać pojedynczo. Najwięcej kłopotów sprawiało programowi odręczne pismo, lecz niemało trudności było też z identyfikacją tekstu drukowanego.

Neisser i Selfridge tak podsumowali wyniki swoich badań oraz wysiłki innych uczonych w tym względzie:

Program zapewnia tylko 10% mniej prawidłowych identyfikacji niż człowiek – jest to z pewnością dobry wynik. Jednocześnie zagadnienia, którym program nie jest w stanie podolać, wskazują na problemy oczekujące na rozwiązanie. Można by tu wyróżnić trzy problemy ogólne: segmentyzacja, uczenie się hierarchiczne oraz generacja cech. (...) Najważniejszy problem uczenia się jest ciągle nietknięty: żaden z obecnych programów nie generuje własnych cech dla przeprowadzenia testów.<sup>30</sup>

W dalszych pracach uwidoczniły się propozycje rozwiązań wspomnianych problemów, czego przykładem są badania L. Uhra i Ch. Vosslera.

W programie L. Uhra i Ch. Vosslera<sup>31</sup> obraz był przekazywany komputerowi jako 20×20-elementowa macierz zer i jedynek. Pusta kratka była zaznaczana zerem, a pełna – jedyneką. Program, za pomocą jednej z kilku metod opartych na przypadkowości, generował i zestawiał operatory czyli charakterystyczne cechy danej litery. Następnie stosował ten zbiór operatorów do transformacji nieznanej macierzy wejściowej i porównywał z wykazem cech zawartych w pamięci. Każda z nich odpowiadała jednej klasie uprzednio badanych liter. Test prawdopodobieństwa wybierał listę najbardziej zbliżoną do listy obliczonych cech, a jej nazwa była wybierana jako nazwa obrazu wejściowego.

Operatory były sprawdzane w zależności od tego, czy przyczyniły się do rozpoznania obrazu czy też nie. Powodowało to wyeliminowanie operatorów dających złe charakterystyki. Możliwe też było rozpoczęcie działania programu bez operatorów; były one generowane, aż do osiągnięcia pewnego pułapu, a wtedy nieodpowiednie operatory były eliminowane przez nowe; prowadziło to do wypracowania optymalnego zbioru operatorów, do przetwarzania danych wejściowych. Program Uhra i Vosslera o 2 tysiącach rozkazów uruchomiony na komputerze IBM-709 rozpoznawał pojedynczą literę w ok. 60 sekund. Lepsze wyniki osiągnęli obywaj naukowcy po licznych poprawkach wprowadzonych do programu. Ulepszona wersja odczytywała poprawnie 96% ręcznie napisanych liter z 26-literowego alfabetu.

Program próbował symulować siatkówkę ludzkiego oka, będącą macierzą równoległych elementów połączonych z wieloma warstwami neuronów.

<sup>30</sup> Tamże s. 250.

<sup>31</sup> Badania prowadzone były od 1959 r., artykuł opublikowany dopiero w 1961 r.: L. U h r, Ch. V o s s l e r, *A pattern-recognition program that generates, evaluates and adjusts its own operators*, Proceedings of the Western Joint Computer Conference 19 (1961) s. 555-570; polski przekład: *Program rozpoznawania obrazów, który wytwarza, ocenia i zmienia swoje operatory*, w: MMiM, s. 265-266.

Budowa żywych układów i metody obróbki sygnałów, jakie w nich występują, w szczególności dotyczy to systemu wzrokowego człowieka, sugerują możliwe rozwiązania maszyn rozpoznających obrazy. Z pewnością nie sugerują rozwiązań jedynych. Nie powinny też być niewolniczo naśladowane.<sup>32</sup>

Uhr i Vossler z dużą dozą optymizmu twierdzili, że:

Program taki zastępuje „programistę analizującego” przez zaprogramowany operator... Tak więc ani programista, ani program nie muszą wiedzieć z góry nic szczególnego o problemie. Jako część swojej normalnej pracy program przeprowadza zbieranie danych, analizę oraz wnioskowanie, co zwykle pozostawia się programiście.<sup>33</sup>

Choć naukowcy przesadzali w swoim optymizmie, Dreyfus zauważa, że względny sukces programu Uhr-Vosslera generującego i zmieniającego swoje własne operatory pokazuje, że podobne problemy są częściowo do rozwiązania. Lecz tego typu rozpoznawanie zależy od ograniczonego zestawu cech ustalanych *ad hoc* czy ogólnych, programowanych czy generowanych. Jak długo mechaniczne rozpoznawanie będzie od nich zależne, badania nie posuną się dalej. „Liczba cech, które mogą być przeglądnięte w rozsądnym czasie, jest ograniczona i obecne programy osiągnęły tę technologiczną granicę” – konkluduje Dreyfus.<sup>34</sup>

Programy z podpoła *Complex-Formal*,<sup>35</sup> bo o nich mowa, rozpoznawały złożone obrazy w szumach poprzez poszukiwanie prawidłowości. Eliminowane były rzeczy niepotrzebne i określany rodzaj obrazu. Dreyfus nie widział natomiast możliwości sformalizowania rozpoznawania urozmaiconych oraz zniekształconych obrazów i tego typu rozpoznawanie ogólne umieścił on w czwartym podpołu *pattern recognition (Nonformal)*; nie przyporządkował mu żadnego rodzaju programu komputerowego.<sup>36</sup>

## 2. Bariery w rozpoznawaniu obrazów

U schyłku lat pięćdziesiątych i na początku sześćdziesiątych percepcyjne możliwości komputerów były ograniczone: przyjmowały na wejściu informacje w postaci bitów i tylko wykonywały wysoce wyspecjalizowane operacje na starannie wprowadzonych danych; nie dokonywały natomiast uogólnień w wysokim stopniu. Człowiek zaś, z powodzi danych pochodzących od zmysłów, potrafi wyabstrahować z nich informacje mające znaczenie dla prowadzonej w danej chwili przez niego działalności.

Dreyfus, analizując problemy związane z ograniczeniami czwartego pola sztucznej inteligencji, powołuje się na poglądy A. Gurwitscha oraz L. Wittgensteina co do koncepcji rozpoznawania oraz podobieństw.

---

<sup>32</sup> Uhr, Vossler, *Program rozpoznawania obrazów, który wytwarza, ocenia i zmienia swoje operatory*, s. 265-266.

<sup>33</sup> Tamże s. 266.

<sup>34</sup> WCCD, s. 98.

<sup>35</sup> WCCD, s. 292-293.

<sup>36</sup> WCCD, s. 292, 294.

Gdy dany przedmiot jest rozpoznawany jako podobny do innych, to zwykle dzieje się to bez świadomości, że jest on przykładem danej kategorii przedmiotów lub elementem określonej klasy specyficznych cech. Pogląd ten zaprezentował A. Gurwitsch w artykule<sup>37</sup> poświęconym analizie różnic zachodzących między świadomością percepcyjną a koncepcyjną. Według Gurwitscha postrzeganie pewnego typu przedmiotu nie jest tym samym, co pojmowanie go jako reprezentacyjnego lub jako szczególny przypadek danego typu. Pierwszym krokiem do utworzenia świadomości koncepcyjnej jest wyodrębnienie charakterystycznych cech danego przedmiotu, które dotąd były nierozdzielnie związane z tą rzeczą. W następstwie tego cechy charakterystyczne stają się cechami ogólnymi i przeciwstawiają się postrzeganej rzeczy, z której zostały wyodrębnione. W ten sposób właściwości te stają się odrębnymi cechami, choć wcześniej wchodziły w skład obrazu, odgrywając w nim drugorzędną rolę. Następuje tu przejście od świadomości percepcyjnej do koncepcyjnej, od odbiorczego do matematycznego stanu umysłu. Percepcyjne rozpoznawanie obrazów uznaje Gurwitsch za ukryte (*implicit*), zaś koncepcyjne za jawne (*explicit*).

Rozróżnienia powyższe posłużyły Gurwitschowi do określenia tzw. ogólnego rozpoznawania (*generic recognition*). Ta forma rozpoznawania, będąc niejawną, nadaje się do wyszczególnienia pewnego zbioru cech. W koncepcji tej pozostaje nierozwiązany problem cech znaczących, a wśród nich rozstrzygających oraz kryterium definiującego.

W odniesieniu do ostatniego zagadnienia Wittgenstein wprowadził dodatkowo pojęcie symptomu, czyli zjawiska zbieżnego w ten czy inny sposób ze zjawiskiem kryterium definiującego. W rozróżnieniu między symptomem a kryterium zasadniczą kwestią dla Wittgensteina<sup>38</sup> jest to, że takie rozróżnienie nie jest ustalone raz na zawsze, dzięki czemu ludzkie koncepcje poznawcze charakteryzują się otwartością i elastycznością.

Wittgenstein rozróżnia rozpoznawanie podobieństwa (*recognition of resemblance*), podobieństwa rodzinne (*family resemblance*) oraz rozpoznawanie zgodności (*recognition of similarity*).

W rozpoznawaniu podobieństwa (*recognition of resemblance*) kontekst odgrywa podstawową rolę. Może on doprowadzić do zauważenia cech, które następnie będą również zauważane w izolacji od kontekstu (jak w przypadku dwuznacznych figur np. kaczka-królik Wittgensteina) albo pozwala skoncentrować uwagę na pewnych aspektach obrazu (eksperyment Pudowkina). W niektórych przypadkach cechy niezbędne do rozpoznania podobieństwa nie mogą być wyizolowane i zdefiniowane w neutralny i bezkontekstowy sposób; kontekst nie tylko określa właściwości istotne dla rozpoznania, lecz kontekst jest określany również przez właściwości.

<sup>37</sup> A. Gurwitsch, *On the Conceptual Consciousness*, w: *The Modeling of Mind*, ed. K.M. Sayre, F.J. Crosson, South Bend: Notre Dame University Press 1963, s. 203.

<sup>38</sup> L. Wittgenstein, *Philosophical Investigations*, Oxford: Blackwell 1953, s. 25.

W innym przypadku podobieństwa, obiekty rozpoznawane jako należące do siebie, wcale nie potrzebują posiadać żadnych wspólnych cech, nawet cech zależnych od kontekstu.

Widzimy skomplikowaną sieć podobieństw nakładających się i przeplatających wzajemnie: czasami są to ogólne podobieństwa, czasami szczegółowe. Nie przychodzi mi na myśl lepsze określenie dla tych podobieństw niż „podobieństwa rodzinne” (*family resemblance*), ponieważ różne podobieństwa wśród członków rodziny: budowa, właściwości, kolor oczu, sposób chodzenia, charakter itd., itd., nakładają się na siebie i przeplatają wzajemnie w ten sam sposób.<sup>39</sup>

Podobieństwo rodzinne jest rozpoznawane jedynie w stosunku do rzeczywistych lub wyobraźalnych przykładów zarówno typowych jak i nietypowych. Przykładowo, nietypowy członek rodziny może zostać rozpoznany przez umieszczenie go w szeregu twarzy prowadzącym od typowego członka do nietypowego. Ten rodzaj rozpoznawania członka rodziny jest osiąganym nie za pomocą zestawienia cech, lecz przez zobaczenie przypadku, o który chodzi; takie rozpoznawanie daje jeszcze jeden rodzaj otwartości i elastyczności. Ci, którzy są zdolni do rozpoznawania członka „rodziny”, nie muszą wymieniać jakichkolwiek cech wspólnych nawet dla dwóch członków, nie ma również żadnego powodu, aby przypuszczać, że takie cechy istnieją.

Wittgenstein sugeruje też trzeci rodzaj rozpoznawania – rozpoznawanie zgodności<sup>40</sup> (*recognition of similarity*), w którym może nie być żadnych wspólnych cech, nawet cech nakładających się; ten rodzaj rozpoznawania w sposób niezbyt jasny wyróżnia on z podobieństwa (*recognition of resemblance*). Bardziej spójnym sposobem zrozumienia jego analizy byłoby wyciągnięcie wniosku, że każda z cech, które wymienia przy omawianiu podobieństwa rodzinnego: budowa, kolor oczu, chód, itp. nie jest identyczna u żadnego z dwóch członków rodziny, lecz okazuje się być złożona z sieci przeplatających się podobieństw. Według tej analogii, każde włókno jest złożone z nowych włókien; w ten sposób dwaj członkowie rodziny nie muszą posiadać żadnych wspólnych cech, aby dzielić podobieństwo.

Autor *What Computers Can't Do* akceptuje poglądy Wittgensteina co do rozpoznawania podobieństw. Dreyfus uważa, że Wittgensteinowskie idee dobrze odzwierciedlają otwartość i elastyczność ludzkiego rozpoznawania obrazów. Komputery są pozbawione takiej otwartości i elastyczności. Podobieństwo w analizach Wittgensteina jest pojęciem podstawowym i nie może zostać zredukowane do wykazu określonych właściwości, a tego właśnie wymagałoby programowanie. Sformalizowanie podobieństwa rodzinnego przez podanie wykazu podobnych cech, wyeliminowałoby pewien typ otwartości na nowe przypadki, co jest przecież najbardziej uderzającą właściwością tej formy rozpoznawania.

Dreyfus odwołuje się również do prezentowanych już poprzednio form „przetwarzania informacji”: pobocznej świadomości, intuicji oraz zależności kontekstowej i za Wittgensteinem nazywa kombinację tych trzech elementów „wyraźnym grupowaniem” (*perspicuous grouping*); uważa je za tak samo ważne jak trzy

<sup>39</sup> Tamże s. 32.

<sup>40</sup> Tamże s. 32.

fundamentalne formy przetwarzania informacji, z których powstało. Wyraźne grupowanie jest specyficznie ludzką formą przetwarzania informacji.

Komputer, by rozpoznawać obrazy, musi koncepcyjnie określić przynależność danego obrazu do pewnej klasy; robi to przez analizowanie listy cech, co grozi wykładniczym wzrostem. Człowiek nie musi tworzyć takiej koncepcji i wykazu wspólnych cech – dzięki kontekstowi redukuje on wieloznaczność i nie traktuje rozpoznawania obrazów jako pewnego rodzaju klasyfikacji, unika też wzrostu wykładniczego przez specyficzne ludzkie przetwarzanie informacji. Mechaniczne techniki dały sukcesy na początku prac związanych z formalizowaniem; problemy rozpoczęły się przy trudniejszych zagadnieniach związanych z rozpoznawaniem ludzkiej twarzy czy form artystycznych. Za słaby punkt programów uważa Dreyfus to, że nie mogą one określić własnych operatorów wyboru, z czym wiążą się inne problemy.

Tolerancja zmian jest jedną ze wskazówek mówiących o różnicy między komputerowym a ludzkim rozpoznawaniem obrazów. Różnica ta jest znaczna i dotyczy orientacji przestrzennej, rozmiarów, stopnia niekompletności i zniekształcenia oraz poziomu szumu tła. Początkowo naukowcy próbowali przy pomocy programu normalizować zbiór, by potem porównać go ze zbiorem matryc i odnaleźć zgodność obrazu z matrycą. Człowiek natomiast nie musi wcale normalizować obrazu – może odbierać go jako duży, mały, niekompletny etc. i równie dobrze go rozpoznawać. Ludzkie rozpoznawanie toleruje takie zmiany. Następna generacja programów nie normalizowała już obrazu, ale dążyła do określenia mocnych operatorów, które szukałyby wyróżniających cech i były nieczułe na szumy oraz zniekształcenia. Lecz jak się wydaje, człowiek również nie stosuje takich sposobów w nawet trudnych przypadkach. Gdy ucieka się on do słownych wskazówek, nie są one mocnymi operatorami akceptującymi rozmyte obrazy i pomijającymi szumy; są raczej zbiorem idealnych cech będących jedynie przybliżeniem w konkretnym obrazie. Obrazy zniekształcone są rozpoznawane jako przedstawiające te same cechy, co figury bez zniekształceń, lecz z pewnymi przypadkowymi dodatkami lub brakami. Podobnie jest z szumem – nie jest on badany i oddzielany, lecz pomijany jako nieistotny. W tym wypadku ponownie daje znać o sobie ludzka zdolność do rozróżniania elementów istotnych od nieistotnych.

Uczeni proponowali dwie metody w celu określenia, który z już przeanalizowanych obrazów najbardziej przypomina matrycę. Pierwszą metodą była analiza przedstawionego obrazu poprzez drzewo decyzji (*decision tree*), drugą łączenie prawdopodobieństw występowania zbiorów cech, jak robi to program *Pandemonium* Selfridge'a. Według Dreyfusa jedna i druga metoda bezkrytycznie zakłada, że istota ludzka, tak jak automatyczny program rozpoznający obrazy, musi sklasyfikować specyficzną listę cech obrazu, a tak właśnie sądzą Selfidge i Neisser: „człowiek oddzielający wzór z kompleksu bodźców istotnie sklasyfikował możliwe dane wejściowe”<sup>41</sup>

<sup>41</sup> Tamże s. 238.



Earl Hunt robi takie same założenie w przeglądzie prac nad rozpoznawaniem obrazów: „Rozpoznawanie obrazów, tak jak uczenie się znaczenia, implikuje naukę pewnej reguły klasyfikacji”<sup>42</sup>

Lecz w tym wypadku Dreyfus zauważa niebezpieczeństwo wzrostu wykładniczego, gdy obraz jest skomplikowany i na tyle podobny do innych, że trzeba rozważać wiele cech, by go rozpoznać. Można przyjąć wtedy metodę analizowania kolejnych cech, by obraz został rozpoznany, ale to prowadzi do założenia istnienia pewnych decydujących cech, które sprawiają, że rozpoznawanie jest w ogóle wykonalne. Trzeba zatem odnaleźć percepcyjną heurystykę mocnych operatorów i wbudować ją do programu, lecz „jak dotąd nikt nie był w stanie jej odnaleźć”<sup>43</sup> Podobnie zresztą w przypadku programów szachowych, mistrzowie nie są zdolni określić dla programu metod heurystycznych. Na nic zdaje się obrona Selfidga i Neissera przy pomocy założenia, że często nawet samemu analizującemu nie jest znana podstawa klasyfikacji, gdyż jest zbyt złożona, by być jawnie określoną i lista cech jest badana podświadomie.<sup>44</sup> Jak się jednak wydaje człowiek nie analizuje szeregowo lub równoległe wszystkich istotnych cech, by rozpoznać obraz; wiele decydujących cech nigdy nie jest badanych jawnie – one wykonują swoje zadanie pozostając na obrzeżu świadomości. Świadczyłyby o tym trudności związane ze znalezieniem takiego wykazu istotnych cech i problemy ze sformalizowaniem go; te kłopoty ujawniają kolejną barierę dla komputerów w rozpoznawaniu obrazów.

Człowiek rozpoznaje określony obraz jako podobny do innych bez świadomości, że jest on przykładem danego wzoru lub jednym z elementów pewnej klasy zdefiniowanej według specyficznych cech. Tu właśnie uwidacznia się różnica między świadomością percepcyjną a koncepcyjną, o której pisał A. Gurwitsch. Człowiek nie potrzebuje tworzyć koncepcji lub grupować wspólnych cech kilku przykładów tego samego obrazu, by go rozpoznać. To odróżnia ludzkie rozpoznawanie od komputerowego, zachodzącego jedynie na jawnym poziomie koncepcyjnym. Dreyfus przypuszcza, że w ludzkim *pattern recognition*: „przechodzenie od ukrytego percepcyjnego grupowania do jawnej koncepcyjnej klasyfikacji nawet na pewnym etapie końcowym, jak w szachach, jest zwykle niekorzystne”<sup>45</sup>

W przypadku ludzi tylko niektórzy chorzy na afazję utracili zdolność percepcyjnego rozpoznawania. Dla nich ocena tego, co widzą staje się wyłącznie kwestią klasyfikacji, co czyni ich zupełnie niezdolnymi do radzenia sobie w normalnym świecie. Podobnie jak komputer, muszą oni odwoływać się do procedur poszukiwania według wykazu cech. Takie rozpoznanie koncepcyjne jest czasochłonne i nieporęczne.

<sup>42</sup> Tamże s. 246.

<sup>43</sup> WCCD, s. 122.

<sup>44</sup> WCCD, s. 238.

<sup>45</sup> WCCD, s. 123.

### 3. Podsumowanie

Dreyfus sugeruje, by „stanać ponad ideą, podzielaną zarówno przez tradycyjnych filozofów jak i badaczy sztucznej inteligencji, że rozpoznawanie obrazów może zawsze być zrozumiane jako pewien rodzaj klasyfikacji”<sup>46</sup> Poszczególne typy rozpoznawania obrazów zbyt pośpiesznie są uznawane za jeden i żaden z nich nie posiada odpowiedniej właściwości wymaganej przez filozofów i maszyny cyfrowe. Rozpoznawanie nazywane przez Gurwitscha typowym, będąc niejawnym, nadaje się do ujawniania pewnego zbioru cech. Wydaje się więc być odpowiednie do zaprogramowania. Lecz według Dreyfusa, Gurwitsch przeoczył to, że w rozpoznawaniu typowym człowiek wyróżnia cechy znaczące a wśród nich pewne decydujące właściwości. Klasyfikacja zaproponowana przez Wittgensteina wydaje się lepiej odzwierciedlać ludzkie rozpoznawanie obrazów. Podkreśla ona znaczenie kontekstu, zespołu podobieństw rodzinnych nakładających się na siebie wzajemnie nawet bez kontekstu (*family resemblance*), oraz ludzką umiejętność rozpoznawania zgodności, nawet bez żadnych wspólnych cech (*recognition of similarity*). Wittgenstein lepiej oddaje elastyczne i otwarte ludzkie rozpoznawanie obrazów, które stoi w opozycji do sztywnego klasyfikowania przez program komputerowy według ustalonego zbioru cech – podobieństwo w analizach Wittgensteinowskich jest nieformalizowalne.

Dreyfus utrzymuje, że *pattern recognition* wykorzystuje specjalną kombinację trzech form „przetwarzania informacji”: obrzeże świadomości, intuicję oraz zależność kontekstową. Na początku proces jest niejawny – używa on informacji leżącej na obrzeżu świadomości. Aby zrozumieć rolę intuicji, trzeba najpierw odróżnić to co ogólne, od tego co charakterystyczne. Rozpoznawanie rzeczy ogólnych zależy od niejawnych wskazówek, które zawsze można uczynić jawnymi. Rozpoznawanie rzeczy charakterystycznych zależy od podobieństw, których nie można dokładnie określić i tu wielką rolę odgrywa intuicja. W końcu rozpoznawanie, w znaczeniu przybliżania się do wzorca, jest formą zależności kontekstowej.

„Wyraźnym grupowaniem” nazywa Dreyfus, za Wittgensteinem, związek obrzeży świadomości, intuicji oraz kontekstu; grupowanie to uważa autor *What Computers Can't Do* za tak samo ważną formę przetwarzania informacji właściwej człowiekowi, jak trzy składowe, z których powstało. Człowiek jest zdolny do rozpoznawania obrazów, nawet wtedy, gdy są one niekompletne, zdeformowane i obciążone szumami. Cechy wymagane do rozpoznania mogą być tak subtelne i tak liczne, że nawet jeśli można je było sformalizować, to przeszukiwanie listy odgałęzień takich cech, po dodaniu nowych wzorów do wyróżnienia, stałoby się wkrótce niewykonalne. Cechy mogą być zależne od wewnętrznego lub zewnętrznego kontekstu, w ten sposób nie poddają się klasyfikacji bezkontekstowej. Może też nie być wspólnych cech lecz, jak to określił Wittgenstein, „skomplikowana sieć wzajemnie nakładających się podobieństw” zdolnych do przyłączenia wciąż nowych kombinacji.

<sup>46</sup> WCCD, s. 123.

Dowolny system, który ma dorównać ludzkiemu przetwarzaniu, musiałby korzystać z kontekstu i być zdolny do rozpoznawania właściwości istotnych od nieistotnych w danym obrazie; musiałby też używać wskazówek pozostających na obrzeżach świadomości i odbierać pojedynczą rzecz jako typową, tj. ustalić odniesienie rzeczy indywidualnej do modelu.

W złożoności zagadnienia rozpoznawania obrazów należy dopatrywać się przyczyny niewielkiego postępu w badaniach; w czasach, gdy pisana była książka *What Computers Can't Do*, badania te nie wykroczyły poza rozpoznawanie przez komputer prostych obrazów alfanumerycznych, krojów czcionek maszyn do pisania oraz cyfr kodu pocztowego. W 1968 r. Laveen Kanal i B. Chandrasekaran wyrazili następującą opinię:

Ci, którzy mieli nadzieję na model ludzkich procesów rozpoznawania znaleźli się w impasie. Jest prawdopodobne, że te problemy, które dla inżynierów były za trudne, by się z nimi uporać, są dokładnie tymi, jakich należało oczekiwać w bardziej dokładnym rozumieniu systemów ludzkiego rozpoznawania. W każdym razie te odczucia kryzysu są wewnętrznie powiązane z tymi i innymi aspektami sztucznej inteligencji: gram i automatycznym tłumaczeniem.<sup>47</sup>

Dalszy postęp w badaniach nad pozostałymi trzema polami AI (programowaniem gier, rozwiązywaniem problemów, tłumaczeniem języka naturalnego) oczekiwał na przełom w badaniach nad rozpoznawaniem obrazów.

THE LIMITATIONS OF ARTIFICIAL INTELLIGENCE  
BY HUBERT L. DREYFUS (PART II)

S u m m a r y

In this second part of my paper *Limitations of artificial intelligence by Hubert L. Dreyfus* is going about language translation and pattern recognition.

In language translation, even the elements to be manipulated are not clear due to the intrinsic ambiguities of a natural language. Dreyfus set ambiguity tolerance against context-free precision. Word-by-word translation (*mechanical dictionary*) is possible for any computer but translating a natural language (understanding in context of use) is impossible. The order of the words in a sentence does not provide enough information to enable a machine to determine which of several possible parsings is the appropriate one. Also Bar-Hillel and Oettinger, two of the most respected workers in the field of automatic language translation, agree in their pessimistic conclusions concerning the possibility of further progress in the field.

In pattern recognition, difficulties are inextricably intertwined, as well as the fact that similarity and typicality seem to be irreducible characteristics of perception (perspicuous grouping vs. character lists). These difficulties have brought to a standstill the first five years of work on Cognitive Simulation. A computer must recognize all patterns in terms of a list specific traits. This raises of exponential growth which human beings are able to avoid by proceeding in a different way. Any system which can equal human performance, must be able to distinguish the essential features of a particular instance of a pattern and take account of the context. Work in pattern recognition has had a late start and an early stagnation.

---

<sup>47</sup> Cytat za WCCD, s. 98-99.

Indeterminate needs and goals and the experience of gratification which guides their determination cannot be simulated on a digital machine whose only mode of existence is a series of determinate states. It is just because these needs are never completely determined for the individual and for mankind as a whole that they are capable of being made more determinate, and human nature can be retroactively changed by individual and cultural revolutions.