Pionierski specjalistyczny sprzęt

W części 1 odkryłeś, że jednym z powodów niepowodzenia wczesnych wysiłków AI był brak odpowiedniego sprzętu. Sprzęt po prostu nie był w stanie wykonywać zadań wystarczająco szybko, by sprostać nawet przyziemnym potrzebom, a tym bardziej tak złożonym, jak symulowanie ludzkiej myśli. Zagadnienie to zostało opisane w pewnym momencie w filmie The Imitation Game , w którym Alan Turing w końcu złamał kod Enigmy, sprytnie szukając określonej frazy „Heil Hitler" w każdej wiadomości. Bez tej szczególnej wady w sposobie, w jaki operatorzy używali Enigmy, sprzęt komputerowy, z którego korzystał Turing, nigdy nie działałby wystarczająco szybko, aby rozwiązać problem (a ruch miał niemałą przyczepność do sprawy). Co więcej, relacja historyczna - z czego niewiele jest w pełni odtajniona - pokazuje, że problemy Turinga były głębsze niż w filmie. Na szczęście standardowy, gotowy sprzęt może przezwyciężyć problem szybkości wielu problemów dzisiaj i od tego momentu rozpoczyna się ten rozdział. Aby naprawdę zacząć symulować ludzką myśl, potrzebny jest specjalistyczny sprzęt, a nawet najlepszy specjalistyczny sprzęt nie jest dziś do tego zadania. Prawie cały standardowy sprzęt opiera się na architekturze Von Neumann, która oddziela pamięć od obliczeń, tworząc cudownie ogólne środowisko przetwarzania, które po prostu nie działa dobrze w przypadku niektórych rodzajów algorytmów, ponieważ prędkość magistrali między procesorem a pamięcią tworzy wąskie gardło Von Neumanna. Druga część tej części pomaga zrozumieć różne metody stosowane w celu przezwyciężenia wąskiego gardła von Neumanna, dzięki czemu złożone algorytmy wymagające dużej ilości danych działają szybciej. Nawet z niestandardowym sprzętem specjalnie zaprojektowanym do przyspieszania obliczeń, maszyna zaprojektowana do symulacji ludzkiej myśli może działać tak szybko, jak pozwalają na to wejścia i wyjścia. W związku z tym ludzie pracują nad stworzeniem lepszego środowiska, w którym sprzęt może działać. Potrzeba ta może zostać rozwiązana na wiele sposobów, ale tu omówiono dwa: zwiększenie możliwości podstawowego sprzętu i użycie specjalistycznych czujników. Te zmiany w środowisku sprzętowym działają dobrze, ale jak wyjaśnia poniższy materiał, nadal nie wystarczy aby zbudować ludzki mózg. W końcu sprzęt jest bezużyteczny, nawet z ulepszeniami, jeśli ludzie, którzy na nim polegają, nie mogą skutecznie z nim współpracować. Ostatnia część opisuje techniki zwiększania wydajności tych interakcji. Te interakcje są po prostu wynikiem połączenia ulepszonej wydajności i sprytnego programowania. Tak jak Alan Turning wykorzystał sztuczkę, aby jego komputer najwyraźniej zrobił więcej, niż był w stanie zrobić, dzięki tym technikom nowoczesne komputery wyglądają jak cudowni. W rzeczywistości komputer nic nie rozumie; cały zaszczyt przypada osobom, które programują komputer.

Poleganie na standardowym sprzęcie

Większość tworzonych projektów sztucznej inteligencji zaczyna się przynajmniej od standardowego sprzętu, ponieważ nowoczesne gotowe komponenty faktycznie zapewniają znaczącą moc przetwarzania, zwłaszcza w porównaniu z komponentami z lat 80., kiedy AI zaczęła generować użyteczne wyniki. W rezultacie, nawet jeśli ostatecznie nie możesz wykonać pracy na poziomie produkcyjnym przy użyciu standardowego sprzętu, możesz dostać się wystarczająco daleko wraz ze swoim eksperymentalnym i przedprodukcyjnym kodem, aby stworzyć działający model, który ostatecznie przetworzy pełny zestaw danych.

Zrozumienie standardowego sprzętu

Architektura (struktura) standardowego komputera nie zmieniła się od czasu, gdy John von Neumann po raz pierwszy zaproponował go w 1946 roku. Przegląd historii pokazuje, że procesor łączy się z pamięcią i urządzeniami peryferyjnymi za pośrednictwem magistrali w produktach komputerowych już w 1981 r. (I na długo przedtem). Wszystkie te systemy wykorzystują architekturę Von Neumann, ponieważ architektura ta zapewnia znaczne korzyści w zakresie modułowości. Czytanie historii

informuje, że urządzenia te umożliwiają aktualizację każdego komponentu jako indywidualne decyzje, co pozwala na zwiększenie możliwości. Na przykład, w ramach limitów, możesz zwiększyć ilość pamięci lub miejsca dostępnego na dowolnym komputerze. Możesz także użyć zaawansowanych urządzeń peryferyjnych. Wszystkie te elementy łączą się jednak za pośrednictwem magistrali. To, że komputer staje się bardziej wydajny, nie zmienia faktów jego podstawowej architektury. Tak więc komputer, którego używasz dzisiaj, ma taką samą architekturę jak urządzenia utworzone dawno temu; są po prostu bardziej zdolne. Ponadto wielkość urządzenia nie wpływa również na jego architekturę. Komputery w twoim samochodzie polegają na systemie magistrali, który łączy się bezpośrednio z architekturą Von Neumann. (Nawet jeśli rodzaj magistrali jest inny, architektura jest taka sama.) Aby nie myśleć, że jakiekolwiek urządzenie pozostaje nienaruszone, spójrz na schemat blokowy Blackberry. To również zależy od konfiguracji von Neumanna. W związku z tym prawie każde urządzenie, jakie można dziś sobie wyobrazić, ma podobną architekturę, pomimo różnych form, typów magistrali i podstawowych możliwości.

Opisanie standardowych braków sprzętowych

Możliwość stworzenia systemu modułowego ma znaczące zalety, szczególnie w biznesie. Zdolność do usuwania i wymiany poszczególnych komponentów utrzymuje niskie koszty, jednocześnie umożliwiając stopniową poprawę zarówno szybkości, jak i wydajności. Jednak, jak w przypadku większości rzeczy, nie ma bezpłatnego lunchu. Modułowość zapewniana przez architekturę Von Neumann ma kilka poważnych wad:

* Wąskie gardło von Neumanna: Ze wszystkich braków wąskie gardło von Neumanna jest najpoważniejsze, jeśli wziąć pod uwagę wymagania dyscyplin, takich jak sztuczna inteligencja, uczenie maszynowe, a nawet nauka danych.

* Pojedyncze punkty awarii: Każda utrata łączności z magistralą oznacza, że komputer natychmiast ulega awarii. Nawet w systemach z wieloma procesorami utrata pojedynczego procesu, który powinien po prostu spowodować utratę zdolności, zamiast tego powoduje całkowitą awarię systemu. Ten sam problem występuje w przypadku utraty innych komponentów systemu: Zamiast zmniejszać funkcjonalność, cały system ulega awarii. Biorąc pod uwagę, że sztuczna inteligencja często wymaga ciągłego działania systemu, potencjalne poważne konsekwencje rosną wraz ze sposobem, w jaki aplikacja polega na sprzęcie.

* Jednomyślność: magistrala Von Neumann może albo pobrać instrukcję, albo dane wymagane do wykonania instrukcji, ale nie może wykonać obu tych czynności. W konsekwencji, gdy pobieranie danych wymaga kilku cykli magistrali, procesor pozostaje bezczynny, co zmniejsza jego wydajność zadania AI wymagające intensywnych instrukcji jeszcze bardziej.

* Zadanie: Gdy mózg wykonuje zadanie, wiele synaps uruchamia się jednocześnie, umożliwiając jednoczesne wykonywanie wielu operacji. Oryginalny projekt Von Neumann umożliwiał tylko jedną operację na raz i tylko po tym, jak system pobrał zarówno wymagane instrukcje, jak i dane. Obecnie komputery zwykle mają wiele rdzeni, które umożliwiają jednoczesne wykonywanie operacji w każdym rdzeniu. Jednak kod aplikacji musi dokładnie spełniać ten wymóg, więc funkcja często pozostaje nieużywana.

BADANIE RÓŻNICY ARCHITEKTURY HARVARDA

Architekturę Harvarda możesz napotkać podczas badania sprzętu, ponieważ niektóre systemy wykorzystują zmodyfikowaną formę tej architektury w celu przyspieszenia przetwarzania. Zarówno architektura Von Neumanna, jak i architektura Harvarda opierają się na topologii magistrali. Jednak

podczas pracy z systemem Von Neumann sprzęt opiera się na jednej magistrali i pojedynczym obszarze pamięci zarówno dla instrukcji, jak i danych, podczas gdy architektura Harvarda opiera się na poszczególnych magistralach dla instrukcji i danych, i może korzystać z oddzielnych obszarów pamięci fizycznej. Zastosowanie poszczególnych magistral umożliwia systemowi Harvard Architecture pobranie następnej instrukcji podczas oczekiwania na dane z pamięci dla bieżącej instrukcji, dzięki czemu architektura Harvard jest szybsza i bardziej wydajna. Jednak spada niezawodność, ponieważ teraz masz dwie punkty awarii dla każdej operacji: magistralę instrukcji i magistralę danych. Mikrokontrolery, takie jak te, które zasilają kuchenkę mikrofalową, często wykorzystują architekturę Harvarda. Ponadto możesz go znaleźć w nietypowych miejscach z konkretnego powodu. Zarówno iPhone, jak i Xbox 360 używają zmodyfikowanych wersji Harvard Architecture, które opierają się na pojedynczym obszarze pamięci (a nie na dwóch), ale nadal polegają na osobnych magistralach. Powodem użycia architektury w tym przypadku jest Digital Rights Management (DRM). Możesz ustawić obszar kodu pamięci tylko do odczytu, aby nikt nie mógł go modyfikować ani tworzyć nowych aplikacji bez pozwolenia. Z punktu widzenia AI może to być problematyczne, ponieważ jedną z możliwości AI jest pisanie nowych algorytmów (kodu wykonywalnego) w razie potrzeby, aby poradzić sobie z nieprzewidzianymi sytuacjami. Ponieważ komputery PC rzadko wdrażają architekturę Harvarda w czystej postaci lub jako główną konstrukcję magistrali, architektura Harvarda nie zwraca na siebie uwagi.

Korzystanie z procesorów graficznych

Po utworzeniu prototypowego zestawu do wykonywania zadań wymaganych do symulacji ludzkiej myśli na dany temat, może być potrzebny dodatkowy sprzęt zapewniający wystarczającą moc przetwarzania do pracy z pełnym zestawem danych wymaganym od systemu produkcyjnego. Istnieje wiele sposobów na zapewnienie takiej mocy przetwarzania, ale powszechnym sposobem jest użycie procesorów graficznych (GPU) oprócz centralnego procesora maszyny. W poniższych sekcjach opisano domenę problemów, którą rozwiązuje GPU, co dokładnie oznacza termin GPU i dlaczego GPU przyspiesza przetwarzanie.

ROZWAŻAJĄC MASZYNĘ BOMBE ALANA TURINGA

Maszyna Bombe Alana Turinga nie była żadną formą sztucznej inteligencji. W rzeczywistości nie jest to nawet prawdziwy komputer. Złamał kryptograficzne wiadomości Enigmy i to wszystko. Dała jednak do myślenia Turingowi, co ostatecznie doprowadziło do opracowania zatytułowanego „Computing Machinery and Intelligence", który opublikował w latach 50. XX wieku. Jednak sama Bombe była w rzeczywistości oparty na polskiej maszynie o nazwie Bomba. Chociaż niektóre źródła sugerują, że Alan Turing pracował sam, Bombe zostało wyprodukowane z pomocą wielu osób, zwłaszcza Gordona Welchmana. Turing również nie wyskoczył z próżni, gotowy do złamania niemieckiego szyfrowania. Czas w Princeton spędził z takimi znakomitościami jak Albert Einstein i John von Neumann (który później wymyślił koncepcję oprogramowania komputerowego). Artykuły napisane przez Turinga zainspirowały innych naukowców do eksperymentowania i sprawdzenia, co jest możliwe. Specjalistyczny sprzęt wszelkiego rodzaju będzie się pojawiał tak długo, jak naukowcy będą pisać artykuły, odrzucać pomysły, tworzyć własne pomysły i eksperymentować. Kiedy oglądasz filmy lub inne media, zakładając, że w ogóle są one historycznie dokładne, nie odchodź z poczuciem, że ci ludzie właśnie się obudzili pewnego ranka, ogłosili: „Dzisiaj będę genialny!" i zrobił coś cudownego. Wszystko opiera się na czymś innym, więc historia jest ważna, ponieważ pomaga pokazać ścieżkę, którą podążasz, i oświetla inne obiecujące ścieżki - te, które nie są podążane.

Biorąc pod uwagę wąskie gardło von Neumanna

Wąskie gardło von Neumanna jest naturalnym rezultatem użycia magistrali do przesyłania danych między procesorem, pamięcią, pamięcią długoterminową i urządzeniami peryferyjnymi. Bez względu

na to, jak szybko magistrala wykonuje swoje zadanie, przytłaczanie go - czyli tworzenie wąskiego gardła zmniejszającego prędkość - jest zawsze możliwe. Z biegiem czasu szybkość procesora wciąż rośnie, a ulepszenia pamięci i innych urządzeń koncentrują się na gęstości - możliwości przechowywania większej ilości miejsca na mniejszej przestrzeni. W związku z tym wąskie gardło staje się coraz większym problemem z każdą poprawką, powodując, że procesor spędza dużo czasu bezczynnie. W granicach rozsądku możesz przezwyciężyć niektóre problemy, które otaczają wąskie gardło Von Neumanna i wywołać niewielki, ale zauważalny wzrost prędkości aplikacji. Oto najczęstsze rozwiązania:

* Buforowanie: gdy problemy z uzyskaniem danych z pamięci wystarczająco szybko w architekturze Von Neumann stały się oczywiste, dostawcy sprzętu szybko zareagowali, dodając zlokalizowaną pamięć, która nie wymagała dostępu do magistrali. Ta pamięć pojawia się na zewnątrz procesora, ale jest częścią pakietu procesora. Szybka pamięć podręczna jest jednak droga, więc rozmiary pamięci podręcznej są zwykle niewielkie.

* Buforowanie procesora: Niestety zewnętrzne pamięci podręczne nadal nie zapewniają wystarczającej prędkości. Nawet użycie najszybszej dostępnej pamięci RAM i całkowite odcięcie dostępu do magistrali nie spełnia wymagań procesora dotyczących mocy obliczeniowej. W związku z tym dostawcy zaczęli dodawać pamięć wewnętrzną - pamięć podręczną mniejszą niż pamięć zewnętrzna, ale z jeszcze szybszym dostępem, ponieważ jest częścią procesora.

* Pobieranie wstępne: Problem z pamięciami podręcznymi polega na tym, że przydają się tylko wtedy, gdy zawierają poprawne dane. Niestety trafienia w pamięci podręcznej są niskie w aplikacjach, które używają dużej ilości danych i wykonują wiele różnych czynności

zadania Kolejnym krokiem w celu przyspieszenia działania procesorów jest odgadnięcie, które dane aplikacja będzie potrzebować w następnej kolejności i załadowanie jej do pamięci podręcznej, zanim aplikacja będzie tego wymagać.

* Używanie specjalnej pamięci RAM: Możesz zostać pochowany przez zupę alfabetu RAM, ponieważ istnieje więcej rodzajów pamięci RAM, niż większość ludzi sobie wyobraża. Każdy rodzaj pamięci RAM może rozwiązać przynajmniej część problemu wąskiego gardła von Neumanna i działają one w określonych granicach. W większości przypadków usprawnienia dotyczą szybszego pobierania danych z pamięci i do magistrali. Dwa główne (i wiele drobnych) czynników wpływa na szybkość: szybkość pamięci (szybkość, z jaką pamięć przenosi dane) i opóźnienie (czas potrzebny na zlokalizowanie określonego fragmentu danych)

Podobnie jak w wielu innych obszarach technologii, szum może stać się problemem. Na przykład wielowątkowość, czynność dzielenia aplikacji lub innego zestawu instrukcji na dyskretne jednostki wykonawcze, które procesor może obsłużyć pojedynczo, jest często reklamowana jako sposób na przezwyciężenie wąskiego gardła von Neumanna, ale tak naprawdę nie działa cokolwiek więcej niż dodać narzut (co pogarsza problem). Wielowątkowość jest odpowiedzią na inny problem: zwiększenie wydajności aplikacji. Gdy aplikacja dodaje problemy związane z opóźnieniami do wąskiego gardła Von Neumanna, cały system zwalnia. Wielowątkowość zapewnia, że procesor nie marnuje więcej czasu na użytkownika lub aplikację, ale zamiast tego ma coś do zrobienia przez cały czas. Opóźnienie aplikacji może wystąpić w przypadku dowolnej architektury procesora, nie tylko architektury Von Neumann. Mimo to wszystko, co przyspiesza ogólne działanie aplikacji, jest widoczne dla użytkownika i całego systemu.

Definiowanie GPU

Pierwotnym celem procesora graficznego (GPU) było szybkie przetwarzanie danych obrazu, a następnie wyświetlenie uzyskanego obrazu na ekranie. W początkowej fazie ewolucji komputera procesor wykonał całe przetwarzanie, co oznaczało, że grafika mogła pojawiać się powoli, podczas gdy procesor wykonywał inne zadania. W tym czasie komputer zazwyczaj był wyposażony w kartę graficzną, która ma niewielką moc obliczeniową lub nie ma jej wcale. Jedyne, co robi karta graficzna, to konwersja danych komputerowych do postaci wizualnej. W rzeczywistości użycie tylko jednego procesora okazało się prawie niemożliwe, gdy komputer przeszedł obok wyświetlaczy tekstowych lub wyjątkowo prostej 16-kolorowej grafiki. Jednak procesory graficzne tak naprawdę nie wprowadziły wielu zastosowań komputerowych, dopóki ludzie nie zaczęli potrzebować wyjścia 3D. W tym momencie połączenie procesora i karty graficznej po prostu nie było w stanie tego zrobić. Pierwszym krokiem w tym kierunku były systemy takie jak Hauppauge 4860, które zawierały procesor i specjalny układ graficzny (w tym przypadku 80860) na płycie głównej. 80860 ma tę zaletę, że wykonuje obliczenia niezwykle szybko. Niestety te asynchroniczne systemy wieloprocesorowe nie całkiem spełniają oczekiwania ludzi (chociaż były niesamowicie szybkie jak na ówczesne systemy) i okazały się niezwykle drogie. Poza tym cały problem polegał na pisaniu aplikacji, które zawierały ten drugi (lub kolejny) układ. Dwa układy również współużytkowały pamięć (których było dużo dla tych systemów). Procesor graficzny przenosi przetwarzanie grafiki z płyty głównej na kartę graficzną. Procesor może nakazać GPU wykonanie zadania, a następnie GPU określa najlepszą metodę wykonania niezależnie od procesora. Karta graficzna ma osobną pamięć, a ścieżka danych dla magistrali jest ogromna. Ponadto GPU może uzyskać dostęp do pamięci głównej w celu uzyskania danych potrzebnych do wykonania zadania i opublikowania wyników niezależnie od procesora. W rezultacie ta konfiguracja umożliwia nowoczesne wyświetlacze graficzne. Jednak to, co naprawdę wyróżnia GPU, to fakt, że GPU zazwyczaj zawiera setki rdzeni, w przeciwieństwie do zaledwie kilku rdzeni dla procesora. Mimo że procesor zapewnia większą funkcjonalność, GPU wykonuje obliczenia niezwykle szybko i może jeszcze szybciej przenosić dane z GPU na ekran. Ta zdolność sprawia, że GPU specjalnego przeznaczenia jest kluczowym elementem w dzisiejszych systemach. Biorąc pod uwagę, dlaczego procesory graficzne działają dobrze Podobnie jak w przypadku układu 80860 opisanego w poprzedniej sekcji, obecnie procesory graficzne przodują w wykonywaniu specjalistycznych zadań związanych z przetwarzaniem grafiki, w tym pracą z wektorami. Wszystkie rdzenie wykonujące zadania równolegle naprawdę przyspieszają obliczenia AI. W 2011 r. W Google Brain Project padła sztuczna inteligencja, aby rozpoznać różnicę między kotami a ludźmi, oglądając filmy na YouTube. Aby jednak to zadanie zadziałało, Google użył 2000 procesorów w jednym z gigantycznych centrów danych Google. Niewielu ludzi miałoby zasoby potrzebne do powielenia pracy Google. Z drugiej strony Bryan Catanzaro (zespół badawczy NVidii) i Andrew Ng (Stanford) byli w stanie powielić pracę Google'a przy użyciu zestawu 12 procesorów graficznych NVidia. Po tym, jak ludzie zrozumieli, że układy GPU mogą zastąpić wiele systemów komputerowych zaopatrzonych w procesory, mogą zacząć robić różne projekty AI. W 2012 r. Alex Krizhevsky (Uniwersytet w Toronto) wygrał konkurs rozpoznawania obrazów komputerowych ImageNet za pomocą procesorów graficznych. W rzeczywistości wielu naukowców używa teraz układów GPU z niewiarygodnym sukcesem.

Tworzenie specjalistycznego środowiska przetwarzania

Według wielu ekspertów, takich jak Massimiliano Versace, CEO Neurala Inc. (https://www.neurala.com/), głębokie uczenie się i sztuczna inteligencja są procesami innymi niż Von Neumann. Ponieważ zadanie, które wykonuje algorytm, nie jest zgodne z podstawowym sprzętem, istnieją wszelkiego rodzaju nieefektywności, wymagane są ataki hakerskie, a uzyskanie wyniku jest znacznie trudniejsze niż powinno. Dlatego projektowanie sprzętu pasującego do oprogramowania jest dość atrakcyjne. Agencja Obrony Zaawansowanych Projektów Badawczych (DARPA) podjęła jeden taki projekt w postaci Systemów Neuromorficznej Adaptacyjnej Skalowalnej Elektroniki Elektronicznej (SyNAPSE). Ideą tego podejścia jest zduplikowanie podejścia natury do rozwiązywania problemów

poprzez połączenie pamięci i mocy obliczeniowej zamiast rozdzielania tych dwóch elementów. W rzeczywistości zbudowali system (był ogromny) i więcej na ten temat można przeczytać na stronie. Projekt SyNAPSE posunął się naprzód. IBM zbudował mniejszy system, wykorzystując nowoczesną technologię, która była zarówno niezwykle szybka, jak i energooszczędna. Jedynym problemem jest to, że nikt ich nie kupuje. Tak jak wiele osób twierdziło, że Betamax był lepszym sposobem przechowywania danych niż VHS, VHS wygrał pod względem kosztów, łatwości użytkowania i atrakcyjnych funkcji. To samo dotyczy oferty SyNAPSE IBM, TrueNorth. Próby znalezienia ludzi, którzy są gotowi zapłacić wyższą cenę, programiści, którzy mogą opracowywać oprogramowanie przy użyciu nowej architektury, oraz produkty, które naprawdę korzystają z układu, były trudne. W rezultacie kombinacja procesorów i procesorów graficznych, nawet z nieodłącznymi słabościami, nadal wygrywa. W końcu ktoś prawdopodobnie zbuduje układ, który bardziej przypomina biologiczny odpowiednik mózgu. Obecny system który to zrobi prawdopodobnie nie będzie w stanie wytworzyć pożądanego wzrostu mocy obliczeniowej. W rzeczywistości, takie firmy jak Google pracują nad alternatywą taką jak Tensor Processing Unit (TPU), które faktycznie widzi zastosowanie w aplikacjach takich jak wyszukiwarka Google, Street View, GoogleZdjęcia i Tłumacz Google . Ponieważ masz teraz technologię stosowaną w rzeczywistych aplikacjach na dużą skalę, niektórzy ludzie również kupują układy, niektórzy programiści wiedzą, jak pisać dla nich aplikacje, i istnieją przekonujące produkty, których ludzie wymagają. W przeciwieństwie do SyNAPSE, TPU opiera się również na dobrze znanej technologii ASIC (Application Specific Integrated Circuit), która znalazła zastosowanie w niezliczonych aplikacjach, więc to, co naprawdę robi Google, to zmiana istniejącej technologii. W rezultacie szanse na sukces tego typu układów na rynku są znacznie wyższe niż w przypadku SyNAPSE, który opiera się na całkowicie nowej technologii.

Zwiększanie możliwości sprzętowych Procesora nadal działa dobrze w systemach biznesowych lub w aplikacjach, w których potrzeba ogólnej elastyczności programowania przewyższa czystą moc przetwarzania. Jednak procesory graficzne są obecnie standardem dla różnych rodzajów danych, uczenia maszynowego, sztucznej inteligencji i potrzeb głębokiego uczenia się. Oczywiście, wszyscy ciągle szukają następnej wielkiej rzeczy w środowisku programistycznym. Zarówno procesory, jak i procesory graficzne są procesorami na poziomie produkcyjnym. W przyszłości możesz zobaczyć jeden z dwóch rodzajów procesorów zamiast tych standardów:

* Układy scalone specyficzne dla aplikacji (ASIC): W przeciwieństwie do ogólnych procesorów, sprzedawca tworzy układ ASIC do określonego celu.

Rozwiązanie ASIC oferuje niezwykle szybką wydajność przy niewielkim zużyciu energii, ale brakuje mu elastyczności. Przykładem rozwiązania ASIC jest Google Tensor Processing Unit (TPU), który jest wykorzystywany do przetwarzania mowy.

* Programowalne tablice bramek (FPGA): Podobnie jak w przypadku ASIC, sprzedawca zazwyczaj wytwarza FPGA do określonego celu. Jednak w przeciwieństwie do ASIC, możesz zaprogramować układ FPGA, aby zmienić jego podstawową funkcjonalność. Przykładem rozwiązania FPGA jest Brainwave Microsoftu, który jest wykorzystywany do projektów głębokiego uczenia się. Bitwa między układami ASIC i FPGA zapowiada się gorąco, a deweloperzy AI stają się zwycięzcami. Na razie wydaje się, że Microsoft i FPGA przejęły inicjatywę. Chodzi o to, że technologia jest płynna i należy spodziewać się nowych rozwiązań.

Dostawcy pracują również nad zupełnie nowymi typami przetwarzania, które mogą, ale nie muszą, działać zgodnie z oczekiwaniami. Na przykład Graphcore pracuje nad jednostką przetwarzania danych (IPU). Wiadomość o nowych procesorach musisz zebrać z odrobiną soli, biorąc pod uwagę szum, który

otaczał przemysł w przeszłości. Kiedy zobaczysz prawdziwe aplikacje dużych firm, takich jak Google i Microsoft, możesz poczuć się nieco bardziej pewny przyszłości technologii.

Dodawanie wyspecjalizowanych czujników

Istotnym składnikiem sztucznej inteligencji jest zdolność sztucznej inteligencji do symulacji ludzkiej inteligencji przy użyciu pełnego zestawu zmysłów. Dane wejściowe dostarczane przez zmysły pomagają ludziom rozwijać różne rodzaje inteligencji opisane w części 1. Zmysły ludzkie zapewniają właściwy rodzaj danych wejściowych do stworzenia inteligentnego człowieka. Nawet zakładając, że AI może w pełni wdrożyć wszystkie siedem rodzajów inteligencji, nadal wymaga odpowiedniego rodzaju danych wejściowych, aby ta inteligencja mogła funkcjonować. Ludzie zazwyczaj mają pięć zmysłów do interakcji ze środowiskiem: wzrok, dźwięk, dotyk, smak i słuch. Co dziwne, ludzie nadal nie rozumieją w pełni swoich możliwości, więc nic dziwnego, że komputery opóźniają się, jeśli chodzi o wyczuwanie środowiska w taki sam sposób, jak ludzie. Na przykład do niedawna smak składał się tylko z czterech elementów: soli, słodyczy, gorzkiego i kwaśnego. Jednak na liście pojawiają się teraz jeszcze dwa smaki: umami i tłuszcz. Podobnie niektóre kobiety są tetrachromatami), które widzą 100 000 000 kolorów, a nie więcej niż 1 000 000 (tylko kobiety mogą być tetrachromatami ze względu na wymagania chromosomalne). Wiedza, ile kobiet ma taką możliwość, nie jest jeszcze możliwa. Wykorzystanie filtrowanych danych statycznych i dynamicznych umożliwia dzisiejszej AI interakcję z ludźmi w określony sposób. Weźmy na przykład Alexę , urządzenie Amazon, które najwyraźniej cię słyszy, a następnie coś mówi. Chociaż Alexa nie rozumie nic, co mówisz, wygląd komunikacji jest bardzo uzależniający i zachęca ludzi do antropomorfizacji tych urządzeń. Aby w ogóle wykonać swoje zadanie, Alexa wymaga dostępu do specjalnego czujnika: mikrofonu, który pozwala mu słyszeć. W rzeczywistości Alexa ma wiele mikrofonów, które pomagają słyszeć wystarczająco dobrze, aby zapewnić iluzję zrozumienia. Niestety, choć jest tak zaawansowana, jak Alexa, nie może niczego zobaczyć, poczuć, dotknąć ani posmakować, co czyni go dalekim od człowieka w najmniejszy nawet sposób. W niektórych przypadkach ludzie naprawdę chcą, aby ich AI posiadało nadrzędne lub różne zmysły. AI, która wykrywa ruch w nocy i reaguje na niego, może polegać na podczerwieni, a nie na normalnym widzeniu. W rzeczywistości użycie alternatywnych zmysłów jest obecnie jednym z ważnych zastosowań AI. Możliwość pracy w środowiskach, w których ludzie nie mogą pracować, jest jednym z powodów, dla których niektóre rodzaje robotów stały się tak popularne, ale praca w tych środowiskach często wymaga zestawu nieludzkich czujników. W związku z tym temat czujników dzieli się na dwie kategorie (z których żadna nie jest w pełni zdefiniowana): czujniki podobne do ludzkich i alternatywne czujniki środowiskowe.

Opracowywanie metod interakcji ze środowiskiem

AI, które jest samowystarczalne i nigdy nie wchodzi w interakcje ze środowiskiem, jest bezużyteczne. Oczywiście ta interakcja przybiera formę danych wejściowych i wyjściowych. Tradycyjna metoda dostarczania danych wejściowych i wyjściowych odbywa się bezpośrednio za pośrednictwem strumieni danych, które komputer może zrozumieć, takich jak zestawy danych, zapytania tekstowe i tym podobne. Jednak podejścia te są mało przyjazne dla człowieka i wymagają specjalnych umiejętności w użyciu. Interakcja z AI coraz częściej występuje w sposób, który ludzie rozumieją lepiej niż w przypadku bezpośredniego kontaktu z komputerem. Na przykład dane wejściowe są przekazywane przez szereg mikrofonów, gdy zadajesz pytanie Alexie. AI zamienia słowa kluczowe w pytaniu na tokeny, które może zrozumieć. Te tokeny inicjują następnie obliczenia, które tworzą dane wyjściowe. AI tokenizuje dane wyjściowe w zrozumiałą dla człowieka formę: zdanie mówione. Następnie usłyszysz zdanie, gdy Alexa mówi do ciebie przez mówcę. Krótko mówiąc, aby zapewnić użyteczną funkcjonalność, Alexa musi wchodzić w interakcje ze środowiskiem na dwa różne sposoby, które przemawiają do ludzi, ale których Alexa tak naprawdę nie rozumie. Interakcje mogą przybierać różne formy. W rzeczywistości liczba i

formy interakcji stale rosną. Na przykład AI może teraz wąchać. Jednak komputer tak naprawdę nic nie wącha. Czujniki umożliwiają przekształcenie detekcji chemicznej w dane, które AI może następnie wykorzystać w taki sam sposób, jak wszystkie inne dane. Możliwość wykrywania substancji chemicznych nie jest nowa; możliwość zmiany analizy tych chemikaliów nie jest niczym nowym; algorytmy nie są również używane do interakcji z danymi wynikowymi. Nowością są zestawy danych używane do interpretacji danych przychodzących jako zapachu, a te zbiory danych pochodzą z badań na ludziach. Nos AI ma wiele możliwych zastosowań. Na przykład, pomyśl o zdolności AI do używania nosa podczas pracy w niektórych niebezpiecznych środowiskach, takich jak zapach wycieku gazu, zanim będzie można go zobaczyć za pomocą innych czujników. Wzrastają również interakcje fizyczne. Roboty działające na liniach montażowych to stary kapelusz, ale należy wziąć pod uwagę efekty robotów, które potrafią prowadzić. Są to większe zastosowania interakcji fizycznej. Weź również pod uwagę, że AI może reagować na mniejsze sposoby. Na przykład Hugh Herr wykorzystuje AI do interakcji z inteligentną stopą . Ta dynamiczna stopa stanowi doskonały zamiennik dla osób, które straciły prawdziwą stopę. Zamiast statycznego rodzaju sprzężenia zwrotnego, które człowiek otrzymuje ze standardowej protezy, ta dynamiczna stopa faktycznie zapewnia rodzaj aktywnego sprzężenia zwrotnego, które ludzie są przyzwyczajeni do uzyskiwania z prawdziwej stopy. Na przykład ilość odepchnięcia od stopy różni się podczas chodzenia pod górę niż podczas zejścia. Podobnie, poruszanie się po krawężniku wymaga odmiennej odpowiedzi niż nawigacja o krok. Chodzi o to, że w miarę jak AI staje się bardziej zdolne do wykonywania złożonych obliczeń w mniejszych pakietach z coraz większymi zestawami danych, zwiększa się zdolność AI do wykonywania interesujących zadań. Jednak zadania wykonywane przez sztuczną inteligencję mogą obecnie nie mieć kategorii ludzkiej. Nie możesz nigdy naprawdę wchodzić w interakcje z AI, która rozumie twoją mowę, ale możesz polegać na AI, która pomaga ci utrzymać życie lub przynajmniej uczynić go bardziej znośnym.