

Przez kilka ostatnich stuleci do rozwiązywania problemów w wielu dziedzinach ludzie korzystali z tak zwanego podejścia "Leonardo da Vinci", które wymaga szczegółowego zrozumienia problemu, a następnie matematycznego zaprojektowania rozwiązania. Niestety, w przypadku coraz większej liczby problemów, które należą do rozwiązywania, ludzie nie są w stanie ich zrozumieć, więc takie podejście nie jest możliwe. Problemem jest to, że w dobie komputerów jesteśmy przytłoczeni ogromną ilością danych, które są bardzo trudne do zrozumienia i analitycznego przetwarzania. Istnieją także problemy rodowe czy inżynierskie, które nie mogą być opisane przez równania matematyczne. Te bardzo złożone problemy mogą zostać rozwiązane pod warunkiem, że proces projektowania zostanie zastąpiony przez proces uczenia. Takie podejście pozwoli nam na rozwiązanie złożonych problemów wielowymiarowych, nieliniowych bez konieczności ich zrozumienia oraz konieczności zapisania ich formułami matematycznymi.

Aby lepiej wyjaśnić różnicę między projektowaniem, a uczeniem, weźmy pod uwagę przykład psa. Wydaje się, że człowiek nie błądzi w stanie zaprojektowania psa w wszystkie połączenia nerwowe w jego mózgu. Zamiast wprowadzając złożony proces jego projektowania możemy szkolić psa do naszych potrzeb. Proszę zauważyć, że obecny postęp w technologii pozwala nam już wytwarzać chipy krzemowe zbudowane ze 100 miliardów tranzystorów, które kosztują zaledwie kilkadziesiąt dolarów (neuron z sinusoidalną funkcją aktywacji można zbudować z pary tranzystorów). Te chipy krzemowe osiągnęły tak złożoność, że musimy zacząć uczyć je zamiast projektować ich działanie.

Uczenie powinno być realizowane w następujących krokach: najpierw tworzymy złożony system sztucznej inteligencji z losowymi wartościami wag, a następnie system ten, poprzez ich zmianę, jest uczony tak aby rozwiązać zadany problem. Co jest bardzo istotne, w przeciwieństwie do tradycyjnego projektowania, w celu zmiany logiki działania sieci zazwyczaj nie ma potrzeby zmiany jej topologii, ale tylko potrzeba dostosowania wartości wag. Takie podejście uwalnia nas, jako ludzi, od pełnego zrozumienia problemu, a zazwyczaj pozwala go rozwiązać. Jest to nadrzędny cel realizowanego projektu. Pomocne jego znaczenie może mieć istotny wpływ na wiele dziedzin, gdy pozwoli rozwiązać wiele problemów praktycznych, które są trudne w zrozumieniu.

Podróbniej wiat inteligentnych systemów sztucznych sieci neuronowych (SSN), (Artificial Neural Networks - ANN), które wykazały ogromne możliwości, rozpoczął McCulloch-Pitts w 1943 roku. Ludzie szybko zrozumieli ogromne możliwości SSN, ale niestety nie wiedzieli jak je zaprojektować. Opracowanie algorytmu uczenia Error Back Propagation (EBP) w 1989 roku, był zasadniczym przełomem pozwalającym uczyć sieci typu Multi-Layer Perceptron (MLP). W ten sposób proces projektowania został zastąpiony przez proces uczenia.

Za pomocą algorytmu EBP oraz jego modyfikacji, można było rozwiązać wiele trudnych problemów, za algorytm EBP zapoczątkował wzrost zainteresowania obszarem inteligentnych systemów. Jednak algorytm EBP z jego modyfikacjami okazuje się być bardzo powolny, potrzebuje bardzo dużo czasu, aby rozwiązać problem średniej wielkości. Drugim przełomem, na nieco mniejszą skalę, było wprowadzenie algorytmu uczenia LM (Levenberga Marquardt) dla sieci neuronowych MLP [19,20]. Algorytm LM zwiększa szybkość uczenia o 100, a nawet 1000 razy. Niestety, ze względu na potrzeby obliczeń i składowania większej macierzy jacobianu algorytm LM może być stosowany tylko do stosunkowo małych problemów. Drugim ograniczeniem algorytmu LM jest to, że został on opracowany wyłącznie dla określonej architektury MLP i nie może być używany do bardziej zaawansowanych architektur uczenia z dowolnie połączonymi neuronami. Dopiero niedawno przedstawiono nowy algorytm drugiego rzędu NBN [22], w którym oba braki algorytmu LM zostały rozwiązane.

Tymczasem wielu badaczy sfrustrowanych tradycyjnym uczeniem SSN [11] wykorzystuje metody alternatywne, takie jak Support Vector Machines - SVM [5,6] lub Extreme Learning Machines - ELM [7-9]. Te systemy uczenia dla płytkich architektur są bardzo szybkie i skuteczne, ale w celu rozwiązania problemów musiały zostać użyte sieci z nadmierną liczbą neuronów lub jednostek RBF. W rezultacie sieci były od 10 do 100 razy większe niż potrzeba [27, 28]. Nieefektywne sposoby wykorzystania jednostek (neuronów) w SVM i ELM nie dziwi, biorąc pod uwagę, że podczas uczenia ustawiane były wagi tylko jednego wyjściowego neuronu liniowego, a pozostałe wagi pozostałych setek neuronów były ustawiane losowo (ELM) lub w oparciu o wybrane "support vectors" z uczonego zbioru danych (SVM).

Rozwiązując problemy typu Parity-N, wykazano, że możliwości uczenia architektury rośnie liniowo z szerokością sieci i wykładniczo z głębokością sieci [10, 12, 35]. Dlatego też używanie nowych kompaktowych architektur pozwala rozwiązać te same problemy z 10 do 100 razy mniejszą liczbą neuronów niż w przypadku popularnych architektur MLP. Problem polega jednak na tym, że nikt nie wie, jak skutecznie je uczyć. Zauważmy, że rozwój algorytmów uczenia koncentrował się na uczeniu względnie nieskutecznych systemów MLP. Jednym z nielicznych wyjątków był algorytm NBN [22], który jest zdolny do uczenia architektury arbitralnie połączonych neuronów, ale nawet algorytm NBN ma trudności z uczeniem takich głębokich architektur, ze względu na zjawisko "zanikania gradientu" („vanishing gradient”) [16]. Istnieje stosunkowo dobra metoda uczenia głębokiej architektury MLP przez połączenie wstępnego przetwarzania danych, uczenia pierwszych warstw metodami bez nadzoru oraz uczenia ostatnich warstw metodami nadzorowanymi [30-32]. Takie podejście może jednak nie w pełni wykorzystywać możliwości głębokiej architektury, ponieważ, jak już wspomniano, popularne systemy MLP nie są tak potężne. Inna trudność tego podejścia, to wymóg znacznego zaangażowania ludzi, którzy są zdolni do łączenia wielu różnych technik sztucznej inteligencji do rozwiązania konkretnego problemu. W związku z tym, nie będzie łatwo zastosować technologii uczenia głębokiej architektury w innych dyscyplinach.

Niedawno wykazaliśmy, że możliwe jest stosowanie głębokiej topologii innej niż MLP i możliwe jest efektywne jej uczenie na bazie zaawansowanego gradientowego algorytmu drugiego rzędu. Nasze wstępne wyniki mogą być analizowane przez porównanie rysunków 5 i 10 w tym wniosku, a także na rys. 33 w [12], gdzie te wyniki zostały zaprezentowane. W tym przypadku zastosowali my nasz algorytm NBN opracowany kilka lat temu [21,22]. Jednak, należy również opracować

konstruktywny algorytm dla gł bokiej architektury FCC w nieco bardziej skomplikowany sposób, niż ten, który zrobili my dla płytkiej architektury [24]. Taki konstruktywny algorytm dla gł bokich architektur może mieć fundamentalne znaczenie w rozpowszechnianiu tej technologii nauczania w innych dyscyplinach, ze względu na jego łatwość w użyciu.

Proponowane badania podstawowe pomogą zatem zmienić kierunek rozwiązywania bardzo złożonych problemów, zastępując tradycyjne związane z projektowaniem na rzecz podejścia związanego z uczeniem. W ten sposób można potencjalnie rozwiązać bardzo trudne problemy, które są zbyt skomplikowane, aby być w pełni zrozumiałe dla człowieka. Jeśli stosunkowo prymitywne SSN razem z stosunkowo prostym algorytmem EBP pozwalają przekroczyć możliwości wielu wysoko wykwalifikowanych ludzi, to z proponowanymi lepszymi architekturami oraz lepszym algorytmem uczenia, możemy przemieścić inteligentne systemy na znacznie wyższy poziom.