

Deep learning applications in biomedical engineering - streszczenie

Konrad Duraj

1 Wprowadzenie

1.1 Historia sztucznej inteligencji

Pomimo wielu lat badań, sztuczna inteligencja (SI) nadal jest jedną z najbardziej popularnych i tajemniczych zagadnień w informatyce. Sztuczna inteligencja obejmuje wszystko, od niezwykle inteligentnych maszyn po stosowane w grach planszowych algorytmy. Po konferencji w Dartmouth dziedzina sztucznej inteligencji odnosiła znaczne sukcesy przez prawie dwie dekady. Wczesnym przykładem jest słynny program komputerowy ELIZA, stworzony w latach 1964-1966 autorstwa Josepha Weizenbauma z Massachusetts Institute of Technology [HK19]. ELIZA była botem posługującym się językiem naturalnym, zdolnym do symulacji rozmowy z człowiekiem. W 1970 roku Marvin Minsky udzielił wywiadu do Life Magazine, w którym stwierdził, że maszyna o ogólnej inteligencji przeciętnego człowieka może powstać w ciągu trzech do ośmiu lat [HK19].

1.2 Systemy eksperckie

System ekspercki stara się uchwycić wiedzę ludzkiego eksperta - jednostki, która charakteryzuje się wysokim poziomem wiedzy w wysoce wyspecjalizowanym obszarze. Każdy system ekspercki składa się z kilku kluczowych elementów:

1. Bazy wiedzy - jest konstrukcją reprezentującą wiedzę w danej dziedzinie o badanym zjawisku. Może przyjąć dowolną z następujących form:
 - Hierarchiczna, drzewiasta struktura,
 - Baza relacyjna,
 - Grafy,
 - Baza rozmyta.
2. Inżynierii wiedzy - jest to proces pozyskiwania wiedzy od ludzkich ekspertów i przedstawiania jej w komputerze w sposób umożliwiający wyciągnięcie wniosków.

Jedną z najpopularniejszych metod tworzenia systemu eksperckiego jest wykorzystanie logiki rozmytej.

1.3 Logika rozmyta

Logika rozmyta to matematyczna forma radzenia sobie z niepewnością i nieprecyzyjnością w podejmowaniu decyzji. Po raz pierwszy została wprowadzona przez Lotfiego A. Zadeha, matematyka i informatyka, w 1965 roku [Zad65]. Zadeh doszedł do wniosku, że wiele rzeczywistych problemów jest zbyt skomplikowanych, by można je było wyjaśnić konwencjonalną logiką binarną, która akceptuje tylko wartości prawdziwe lub fałszywe. Zamiast tego zasugerował „logikę rozmytą”, bardziej elastyczną metodę, która akceptuje różne stopnie prawdy i częściową przynależność do kategorii. W zależności od tego, jak dobrze dany przedmiot lub pojęcie pasuje do kilku kategorii, logika rozmyta przypisuje różne stopnie przynależności do tych kategorii. Funkcje, które rozmywają cechę wejściową, nazywane są członkostwem/przynależnością. Najpopularniejsze funkcje przynależności stosowane obecnie są wymienione poniżej:

- Singleton,
- Trójkątna,
- Trapezoidalna,
- Funkcja Gauss'a,
- Funkcja Bell'a.

1.4 Upadek systemów eksperckich

Systemy eksperckie były niegdyś okrzyknięte przełomową technologią, która zrewolucjonizowała sposób, w jaki ludzie rozwiązują złożone problemy. Jednak ich popularność spadła na początku 2000 roku, a dziś są coraz rzadziej używane. Jedną z głównych przyczyn upadku systemów eksperckich jest ich niezdolność do uczenia się z danych; opierają się na z góry ustalonym zbiorze zasad i wiedzy dostarczonej przez eksperta. Tak więc te systemy w najlepszym wypadku będą tak dobre, jak inżynierowie oprogramowania, którzy je stworzyli i nie są w stanie dostarczyć nowej wiedzy w zakresie wiedzy naukowej. Ponadto, nie są w stanie przetwarzać nieustrukturyzowanych danych, takich jak:

- obrazy,
- tekst,
- szeregi czasowe,
- grafy,
- nagrania.

co jeszcze bardziej zmniejszyło ich wykorzystanie. Ogólnie rzecz biorąc, upadek systemów eksperckich można w dużej mierze powiązać z ich ograniczoną zdolnością uczenia się i adaptacji, polegania na ekspertach oraz rozwój bardziej wyrafinowanej technologii.

1.5 Uczenie maszynowe

Systemy eksperckie były pierwszą próbą dostosowania komputerów do rozwiązywania problemów w naszym codziennym życiu. Największą wadą był fakt, że wymagały ludzi na każdym kroku, od zbierania danych, poprzez ekstrakcję cech, aż po podejmowanie decyzji. Naukowcy pracujący w dziedzinie sztucznej inteligencji dostrzegli tę wadę i opracowali szereg rozwiązań, które zautomatyzowałyby co najmniej jeden z tych etapów, mianowicie podejmowanie decyzji. Algorytmy te oczekują szeregu cech opisujących konkretny problem i próbują dostosować zestaw parametrów, a tym samym granicę decyzyjną, którą jest hiperpłaszczyzna dzieląca przestrzeń wejściową na różne regiony w oparciu o daną regułę klasyfikacji. Kiedy takiemu algorytmowi podamy wystarczająco dużo przykładów, te modele statystyczne mogą faktycznie stać się dobrym estymatorem dla przyszłych próbek. Istnieje kilka trybów nauki, takich jak:

- Uczenie nadzorowane - używa zestawu szkoleniowego do uczenia modeli w celu uzyskania pożądanego wyniku. Ten zestaw danych szkoleniowych obejmuje dane wejściowe i prawidłowe dane wyjściowe, które umożliwiają modelowi uczenie się. Algorytm mierzy swoją dokładność za pomocą funkcji straty, dostosowując się do momentu, gdy błąd zostanie wystarczająco zminimalizowany,
- Uczenie nienadzorowane - wykorzystuje algorytmy uczenia maszynowego do analizowania i grupowania nieoznakowanych zestawów danych. Algorytmy te odkrywają ukryte wzorce lub grupy danych bez potrzeby interwencji człowieka,
- Uczenie ze wzmocnieniem - stosuje metodę szkolenia polegającą na nagradzaniu pożądanego zachowań i/lub karaniu niepożądanych. Ogólnie rzecz biorąc, agent uczenia się przez wzmocnienie jest w stanie postrzegać i interpretować swoje otoczenie, podejmować działania i uczyć się metodą prób i błędów.

Istnieje kilka algorytmów dla każdego trybu uczenia się. Najpopularniejsze z nich wymieniono poniżej.

1. Uczenie nadzorowane

- Regresja liniowa,
- Regresja logistyczna,
- Drzewa decyzyjne,
- Las losowy,
- Metody gradientowe,
- Maszyna wektorów wspierających.

2. Uczenie nienadzorowane

- K-Najbliższych-Sąsiadów,
- Mikstury Gaussowskie.

3. Uczenie ze wzmocnieniem

- Q-uczenie,
- Aktor-Krytyk,
- SARSA.

Mimo ogromnego sukcesu, jaki metody uczenia maszynowego przyniosły w dobie internetu i "big data", wciąż nie były one wystarczające do rozwiązania wielu problemów informatycznych. Algorytmy uczenia maszynowego nadal cierpią z powodu niemożności radzenia sobie z danymi w ich surowej postaci (audio, wideo, obrazy, tekst). Co więcej, rozwiązania te nadal opierają się na cechach zapewnianych przez człowieka, a więc przy założeniu, że inżynierowie oprogramowania i informatycy są w stanie stworzyć zestaw cech, który zostanie skutecznie wyodrębniony i będzie przedstawiać istotne informacje dla danego problemu.

1.6 Uczenie głębokie

Standardowe techniki uczenia maszynowego miały ograniczone możliwości przetwarzania naturalnych danych w ich surowej formie. Przez dziesięciolecia budowa systemu rozpoznawania wzorców lub uczenia maszynowego wymagała staranności i znacznej wiedzy specjalistycznej w dziedzinie, aby zaprojektować ekstraktor cech, który przekształciłby surowe dane (takie jak wartości pikseli obrazu) do odpowiedniej reprezentacji lub wektora cech z którego podsystem uczący się, często klasyfikator, może wykrywać lub klasyfikować wzorce na wejściu [LBH15]. W ostatnich latach głębokie uczenie stało się wiodącym podejściem do sztucznej inteligencji. Metody głębokiego uczenia to metody uczenia się za pomocą budowania reprezentacji w hierarchiczny sposób, poprzez komponowanie prostych, ale nieliniowych modułów, z których każdy przekształca reprezentację na jednym poziomie (zaczynając od surowych danych wejściowych) w reprezentację na wyższym, nieco bardziej abstrakcyjnym poziomie. Wraz ze wzrostem ilości danych algorytmy te przewyższyły konwencjonalne modele uczenia maszynowego. Nowoczesne architektury nie osiągnęły jeszcze swoich granic w niektórych zadaniach.

Uczenie głębokie poszło o krok dalej niż tradycyjne metody uczenia maszynowego. To podejście automatyzuje proces ekstrakcji cech i podejmowania decyzji w jednej architekturze. Maszyna generuje wyjście w postaci wektora prawdopodobieństwa, po przedstawieniu przykładowych danych podczas szkolenia. Przed treningiem mało prawdopodobne jest aby kategoria docelowa uzyskała najlepszy wynik ze wszystkich kategorii. Błąd (lub odległość) między wynikami wyjściowymi a pożądanym wzorcem jest obliczana przy użyciu funkcji błędu/straty. Aby zmniejszyć tę niedokładność, maszyna zmienia następnie swoje wewnętrzne, regulowane parametry. Te zmienne, często określane jako wagi, to wartości kontrolujące funkcję przetwarzającą informacje wejściowe w wyjściowe. Takich konfigurowalnych wag mogą być setki milionów w typowym systemie głębokiego uczenia. Aby odpowiednio dostosować wagi sieci neuronowej, algorytm uczący się oblicza gradient, który dla każdej wagi wskazuje, o jaką wartość błąd zwiększyłby/zmniejszyłby się, gdyby waga została zwiększona o niewielką wartość. Wagi

są następnie dostosowywane w kierunku przeciwnym do kierunku gradientu. Obecnie jedną z najpopularniejszych procedur optymalizacyjnych jest stochastyczne opadanie gradientu, które po raz pierwszy zostało sformalizowane w [Rob51]. Po obliczeniu wartości błędu dla podzbioru całego zbioru danych obliczamy gradient funkcji straty względem wszystkich wag sieci neuronowej. Obliczanie sygnału wyjściowego jest często określane jako „propagacja w przód”, a propagacja sygnału błędu przez sieć jest często określana jako „propagacja wsteczna”.

Głębokie uczenie się czyni znaczne postępy w rozwiązywaniu problemów, które dotychczas były nieosiągalne pomimo starań społeczności zajmującej się sztuczną inteligencją. Dzięki sukcesowi w identyfikowaniu złożonych struktur w danych wielowymiarowych może być używane w szerokim zakresie aplikacji naukowych i komercyjnych. Przykłady zastosowań przedstawiono poniżej.

- modelowanie języka naturalnego [BMR⁺20],
- tłumaczenie języka naturalnego [DCLT18],
- generowanie tekstu [BMR⁺20],
- klasyfikacja obrazów [BZK22, YWV⁺22, CND⁺22],
- detekcja elementów na obrazie [WDC⁺22, ZSL22, YLDG22],
- semantyczna segmentacja [CDW⁺22, YZW22],
- przewidywanie szeregów czasowych [ZCZX23],
- rozpoznawanie mowy [OEB⁺19],
- klasyfikacja grafów [ZBE⁺19],
- znajdowanie leków [LLH⁺20]
- i wiele innych.

Nowoczesne sieci neuronowe zapewniają bardzo dobre metryki w kontekście uczenia nadzorowanego. Dodając więcej warstw i więcej jednostek w warstwie, głęboka sieć może reprezentować funkcje o rosnącej złożoności. Dysponując wystarczająco dużym modelem i zestawem oznakowanych próbek treningowych, można zastosować głębokie uczenie do wykonania większości zadań, które wymagają szybkiego przekształcenia danych wejściowych na wektor wyjściowy. Do wprowadzania różnych typów danych (tekst, obraz) istnieje kilka architektur, które mogą być używane oddzielnie lub wspólnie aby rozwiązać określone zadanie.

- Wielowarstwowy perceptron - jest to sieć neuronowa składająca się z wielu warstw połączonych węzłów, z których każda odpowiada za obsługę odrębnej funkcji przetwarzania danych wejściowych. Warstwa wejściowa to pierwsza warstwa, która otrzymuje dane wejściowe. Po przejściu przez szereg ukrytych warstw dane wejściowe docierają ostatecznie do warstwy wyjściowej. Wielowarstwowy perceptron jest skutecznym narzędziem do różnych zastosowań, ponieważ może symulować skomplikowane nieliniowe interakcje między wejściami i wyjściami,
- Sieci spłotowe - przeznaczone do przetwarzania danych o określonej, przypominającej siatkę strukturze, są szczególnym podtypem sieci neuronowych. Dane szeregów czasowych, które można przedstawić jako jednowymiarową siatkę reprezentującą próbki w określonych odstępach czasu, oraz dane obrazowe, które można przedstawić jako dwuwymiarową siatkę pikseli lub siatkę wokseli dla obrazów medycznych 3D (MRI, CT). Nazwa „splotowa sieć neuronowa” wskazuje, że sieć wykorzystuje operację matematyczną zwaną splotem. Splot jest wyspecjalizowanym rodzajem operacji liniowej. Sieci spłotowe to po prostu sieci neuronowe, które wykorzystują splot zamiast ogólnego mnożenia macierzy w co najmniej jednej ze swoich warstw,
- Rekurencyjne sieci neuronowe - ten typ sieci neuronowej został specjalnie zaprojektowany do przetwarzania sekwencji - gdzie przewidywanie przyszłości zależy od dowolnej liczby kroków w przeszłość. Sieci RNN przetwarzają sekwencję wejściową po jednym elemencie na raz, zachowując w swoich jednostkach ukrytych wektor stanu, który zawiera informacje o wszystkich poprzednich elementach sekwencji,

- Transformery - jedną z najnowszych rewolucji w dziedzinie głębokiego uczenia się była architektura transformerów. Jest to głęboka sieć neuronowa, która zastąpiła rekurencyjne połączenia mechanizmem samouwagi (*self-attention*). Samouważność pozwala modelowi uchwycić dłuższe zależności między kolejnymi symbolami. Transformer został po raz pierwszy zaprezentowany jako model *seq2seq*, którego zadaniem było tłumaczenie między językami. Dlatego najpierw został zaprojektowany jako model typu koder-dekoder. Ogólna architektura używa mechanizmu samouwagi oraz w pełni połączonych warstw zarówno dla kodera, jak i dekodera.

2 Aplikacje głębokiego uczenia w aplikacjach inżynierii biomedycznej

Głównym celem tej pracy jest ocena aplikacji, w których głębokie uczenie może okazać się pomocne w dziedzinie inżynierii biomedycznej. Pokazuje również, że głębokie uczenie pomimo bycia potężną technologią ma swoje wady, które są kluczowe, zwłaszcza biorąc pod uwagę możliwe zastosowania w dziedzinie medycznej.

2.1 Przewidywanie toksyczności molekuł z wykorzystaniem głębokiego uczenia

Każdego dnia jesteśmy narażeni na dużą liczbę chemikaliów – poprzez nasze środowisko, żywność, medycynę itp. Wiedza o właściwościach niektórych substancji jest kluczowa dla ochrony naszego organizmu przed działaniem niebezpiecznych czynników. W artykule opisano podejście do klasyfikacji właściwości cząsteczki, a mianowicie jej toksyczności. Stopień, w jakim substancja może zaszkodzić żywemu organizmowi wskazuje na toksyczność. Eksperymentalna ocena toksyczności cząsteczki jest czasochłonna, kosztowna i wymaga specjalistycznego personelu i narzędzi. Głębokie uczenie może być stosowane aby automatycznie ocenić tę właściwość. Do treningu sieci neuronowej wykorzystano zbiór danych o nazwie „SMILES Toxicity”, który jest dostępny na popularnym serwisie data science - ”kaggle.com” [Fan19]. Składał się z 7962 różnych molekuł łącznie, z czego 6998 było nietoksycznych, a 964 toksycznych. Cząsteczki są zakodowane używając formatu „SMILES”. Jest to format oparty na ASCII, który definiuje strukturę cząsteczki - jej atomy, wiązania i łączność w ciągu znaków. Zbiór danych został podzielony na zbiory uczące i testowe w następujący sposób:

1. Zbiór treningowy

- Liczba nietoksycznych molekuł: 6760,
- Liczba toksycznych molekuł: 937.

2. Zbiór testowy

- Liczba nietoksycznych molekuł: 238,
- Liczba toksycznych molekuł: 27.

Przetestowano różne kombinacje architektur modelu i hiperparametrów. Na końcu eksperymentów, model będący hybrydą spłotowej i rekurencyjnej sieci neuronowej osiągnął najlepsze wyniki. Ponieważ nierównowaga klas dla tego konkretnego problemu jest znacząca, wprowadzono dodatkowe techniki mające na celu zapobieganie przeuczeniu i minimalizację fałszywych negatywów, takich jak:

- wagowanie klas,
- cykliczny współczynnik uczenia.

Metody te poprawiły wydajność modelu, który początkowo był drastycznie niezbalansowany przez klasę nietoksyczną. Ostateczny model osiągnął 77% skuteczności na zbiorze testowym zachowując przy tym relatywnie wysoką czułość w porównaniu z innymi eksperymentami i alternatywnymi rozwiązaniami, mianowicie - 53%.

2.2 Rozpoznawanie aktywności kierowców na podstawie jednowymiarowej sieci splotowej

Prowadzenie samochodu to złożona czynność, która obejmuje ruchy całego ciała. Wiele badań dotyczące zachowania kierowców prowadzone są w celu poprawy bezpieczeństwa ruchu drogowego. Ten artykuł jest próbą stworzenia klasyfikatora, który na podstawie danych pochodzących ze scenariuszy związanych z nauką jazdy uzyskanych w rzeczywistym ruchu drogowym dzięki inteligentnym okularom będzie skutecznie rozpoznawać akcję wykonywaną przez kierowców. Badania przeprowadzono w rzeczywistych warunkach drogowych wg Rozdziału 4 ustawy o kierujących pojazdami Rzeczypospolitej Polskiej w dwóch grupach ochotników: dziesięciu doświadczonych kierowców (w wieku od 40 do 68 lat) z co najmniej dziesięcioletnim doświadczeniem w prowadzeniu pojazdów i dziesięciu uczących się kierowców, którzy uczęszczali na lekcje jazdy w lokalnej szkole nauki jazdy (w wieku od 18 do 46 lat). Ostateczny zestaw danych składał się z 520 oznaczonych zapisów elektrookulogramu (EOG), przyspieszenia i żyroskopu, ale w zadaniu klasyfikacyjnym brano pod uwagę tylko sygnały EOG. Nagrania były podzielone na cztery klasy w następujący sposób:

- 120 nagrań opisujących proces parkowania samochodu,
- 120 nagrań opisujących przejazd samochodu przez rondo,
- 160 nagrań opisujących przejazd samochodu przez korek w mieście,
- 120 nagrań opisujących przejazd samochodu przez skrzyżowanie.

Utworzony model jest trójwarstwową, jednowymiarową splotową siecią neuronową, optymalizowaną przy pomocy funkcji entropii krzyżowej. Metryki dla każdej z aktywności przedstawione są w tabeli 2.2.

Kategoria	Precyzja	Czułość	Współczynnik F1
parkowanie	0.98	0.91	0.94
rondo	0.97	0.98	0.97
korek	0.95	0.98	0.96
skrzyżowanie	0.95	0.98	0.96

Aby zatrzymać wzrost ofiar śmiertelnych i wypadków drogowych spowodowanych rosnącą liczbą pojazdów na drodze, należy zmienić paradygmat procesu szkolenia kierowców. Szansa na ocenę percepcji kierowców może ujawnić ważne informacje o ich uważności. Technologie wspomagające kierowcę które są precyzyjne i niedrogie, mogą pomóc w promowaniu bezpiecznej jazdy. Jednak zachowanie w czasie rzeczywistym i monitorowanie warunków jazdy wiąże się z trudnościami technicznymi i koniecznością zwracania uwagi na zdrowie kierowcy, w szczególności zawroty głowy spowodowane wydłużoną jazdą, nagłe zmiany oświetlenia, odbicia od okularów lub niekorzystne warunki drogowe.

2.3 Detekcja uderzeń serca w sygnale sejsmokardiograficznym z wykorzystaniem semantycznej segmentacji

Ponieważ jest uznawany za reprezentatywną miarę funkcji serca, wykrywanie bicia serca jest kluczowym elementem analizy sygnału sercowego. Lokalizacja zespołu QRS na elektrokardiogramie to złoty standard wykrywania bicia serca. Sejsmokardiografia (SCG), która wykorzystuje wibracje do pomiaru serca powoli dorównuje elektrokardiografii jako wiarygodna metoda oceny pracy serca dzięki znacznemu rozwojowi czujników i technologii informacyjno-komunikacyjnych (ICT). Istnieje więc naturalna potrzeba stworzenia nowych algorytmów zdolnych do wyodrębnienia odpowiednich części cyklu sercowego z takich danych. Ten artykuł odpowiada na te potrzeby, opracowując model segmentacji w celu wyodrębnienia prawdopodobnie najważniejszej części, bicia serca. Zbiór danych wykorzystany w tej pracy został nazwany „Combined Measurement of ECG, Breathing and Seismocardiogram”, publicznie dostępne na stronie PhysioNet.org. Zawierał on 60 jednoczesnych nagrań Sygnały EKG (odprowadzenia I i II), sygnały oddechowe i sejsmokardiogramy (na osi z) uzyskane z 20 zdrowych ochotników, którzy nie spali i pozostawali w pozycji leżącej na łóżku. Maski segmentacji zostały wygenerowane przez wyodrębnienie pików R z sygnału EKG, przy użyciu algorytmu Pan-Tompkins. Wygenerowany zestaw danych został podzielony na dane dotyczące treningu, walidacji i testów. Architektura opracowanego

modelu była w stylu U-Net [PT85] oraz zastąpiła standardowy dwuwymiarowy operator splotu jego jednowymiarowym odpowiednikiem. Model oceniono za pomocą wskaźnika Jaccarda i współczynnika F1. Wyniki dla zestawu danych testowych opisano w tabeli 2.3.

Metoda uśredniająca	Jaccard	F1-score
Micro-uśrednianie	0.99	0.99
Macro-uśrednianie	0.97	0.98
Ważone-uśrednianie	0.99	0.99

2.3.1 Semantyczna segmentacja 12-odprowadzeniowego EKG używając rezyduальной sieci U-Net z blokami ścisku i wzbudzenia

Ocena EKG jest prawdopodobnie najczęściej stosowanym sygnałem biomedycznym do wykonywania pomiarów diagnostycznych. Sygnał ten ma swoje odzwierciedlenie w mechanicznej pracy serca i może informować nas o jego przebiegu oraz stanie fizjologicznym tego narządu. Analiza elektrokardiogramu jest złożonym procesem, który wymaga specjalistycznej wiedzy. W tym badaniu wykorzystano zestaw danych dostarczony przez Uniwersytet Lobaczewskiego, dostępny w Physionet.org. W tym zbiorze danych opatrzone adnotacjami następujące elementy sygnałów 12-odprowadzeniowego EKG:

- kompleks QRS,
- załamek T,
- załamek P.

Początkowy zestaw danych został podzielony na trening, walidację i test w następujący sposób: najpierw podzielono dane do zestawu treningowego i testowego z 80% i 20% w oparciu o pacjenta, a następnie wyodrębniono 20% zestawu szkoleniowego do zbioru walidacyjnego. Opracowany model jako architektura typu koder-dekoder. Model został przeszkolony w dwóch różnych wariantach, jeden z blokami wzbudzenia i drugi bez. Aby zmierzyć wydajność tego modelu użyliśmy wskaźnika Jaccard Index. Wyniki przedstawiono w tabli 2.3.1.

Uśredniania	Macro-Jaccard Index	Micro-Jaccard Index	Weighted-Jaccard Index
Bez bloku ścisku-wzbudzenia	0.8	0.86	0.86
Z blokiem ścisku-wzbudzenia	0.87	0.91	0.91

Stworzony model osiąga wysoki zestaw metryk (skuteczność, pole pod powierzchnią krzywej ROC, specyficzność i czułość) niezależnie na wszystkich kanałach sygnałowych. Dzięki temu to rozwiązanie może być stosowane w każdym typie konfiguracji elektrod jako podstawowy algorytm klasyfikacji rytmu serca i zmienności rytmu serca (HRV), oraz wiele innych parametrów wynikających z określonych fragmentów EKG. Model, ze względu na użycie tylko warstw splotowych zamiast mechanizmu samouwagi i architektury transformera może być wydajnie wdrażany na urządzeniach brzegowych.

3 Nowy paradygmat

Systemy głębokiego uczenia w opiece zdrowotnej muszą być przejrzyste i interpretowalne w zachęcający sposób tak aby budowały zaufanie społeczne do algorytmów i modeli, które będą wykorzystywane do podejmowania krytycznych decyzji. Ten problem jest szczególnie ważny w zastosowaniach związanych z opieką zdrowotną, gdzie wyjaśnienie procesu decyzyjnego jest niezbędne do oceny klinicznej. Gdy model głębokiego uczenia generuje prognozę lub stawia diagnozę, ważne jest, aby lekarze byli w stanie zrozumieć uzasadnienie procesu podejmowania decyzji, aby zapewnić najlepszą możliwą opiekę nad pacjentami. Przyjęcie tych systemów w opiece zdrowotnej może ucierpieć, jeśli pojawi się model, którego nie można uzasadnić ani zrozumieć. Może to prowadzić do nieufności i sceptycyzmu w kierunku technologii. W naukach biomedycznych kluczowym jest aby decyzje podejmowane przez sztuczne sieci neuronowe dało się prześledzić wstecz do określonych cech, ich kombinacji i punktów danych w danych treningowych. Jest to istotne dla zidentyfikowania ewentualnych odchyłań lub słabości w modelach oraz dla potwierdzenia ich precyzji i możliwości zastosowania w różnych scenariuszach klinicznych. Bez wytłumaczalności może być trudno zapewnić wymagane bezpieczeństwo.

3.1 Limity uczenia głębokiego

Chociaż głębokie sieci neuronowe osiągnęły spektakularne wyniki, mają wiele wad. Garry Marcus, profesor psychologii i nauk neuronowych na Uniwersytecie Nowojorskim, napisał artykuł na temat problemów, których głębokie uczenie się jeszcze nie rozwiązało [Mar18]. Wskazał kilka ograniczeń, m.in.:

- Są zasobochłonne - sieci neuronowe działają dobrze pod trzema warunkami:
 1. Zbiór danych, na którym został przeszkolony, jest duży i opatrzony adnotacjami,
 2. Zbiór danych, na którym został przeszkolony, jest zróżnicowany,
 3. Architektura modelu jest wystarczająco duża.

Inteligencja ludzi lub zwierząt potrzebuje kilku przykładów, aby rozwiązać konkretne zadanie, dlatego Marcus uważa, że należy włożyć więcej pracy w głębokie uczenie bez nadzoru.

- Nie tworzą one reprezentacji kompozycyjnych - można to lepiej zrozumieć, rozważając zadanie klasyfikacji obrazów. Chociaż architektury, takie jak konwolucyjne sieci neuronowe, tworzą reprezentację cech wyższego poziomu w celu określenia kategorii obiektu, nie dekonstruują takiego obiektu na jego charakterystyczne części,
- Nie są transparentne,
- Wiedza ekspercka nie jest łatwo dodawana do systemu,
- Brak zdolności do manipulacji symbolami,
- Brak zdolności odróżniania korelacji od przyczynowości.

W ostatnich latach nastąpił rozwój tzw. wytłumaczalnej sztucznej inteligencji (XAI). Metody te miały umożliwić twórcom algorytmów głębokiego uczenia uzyskanie wglądu w wewnętrzne działanie sieci neuronowej. Niektóre z popularnych algorytmów to:

- SHAP (*SHapley Additive exPlanations*) - podejście oparte na teorii gier w celu wyjaśnienia wyników dowolnego modelu uczenia maszynowego. SHAP zapewnia ujednoczone podejście do wyjaśnienia danych wyjściowych dowolnego modelu poprzez obliczenie ważności każdej cechy w prognozie,
- GradCAM - działa, wykorzystując gradienty określonej klasy w odniesieniu do końcowej warstwy splotowej, aby wygenerować mapę cieplną ważnych regionów obrazu. Ta technika pomaga nam zrozumieć na których częściach obrazu skupia się model i które cechy obrazu są najważniejsze w prognozowaniu,
- LIME - jest metodą wyjaśniania przewidywań dowolnego modelu czarnej skrzynki poprzez przybliżenie go prostszym modelem.

3.2 Neuronowo-Rozmyte Systemy Inferencji

Zarówno sieci neuronowe, jak i systemy rozmyte mają pewne wady, które prawie całkowicie znikają łącząc oba te zagadnienia. Można je wykorzystać do rozwiązania problemu (np. rozpoznawania wzorców, regresji lub klasyfikacja), jeśli nie istnieje model matematyczny danego problemu. Sieci neuronowe i techniki głębokiego uczenia są szczególnie przydatne w przypadku wielowymiarowych, złożonych oraz nieliniowych danych, w przypadku których wyodrębnianie cech za pomocą ręcznie tworzonych algorytmów, nie jest skalowalne. Biorąc to pod uwagę, można je zastosować wtedy, gdy jest wystarczająca liczba przykładów. Sieci neuronowe są bardzo często pożądanym rozwiązaniem gdyż, z jednej strony nie trzeba podawać wiedzy eksperckiej na temat problemu. Jednak z drugiej strony nie jest łatwo wydobyć zrozumiałe reguły ze struktury sieci neuronowej. Rozmyte systemy wnioskowania stanowią ważną część logiki rozmytej. W większości praktycznych zastosowań (tj. teorii sterowania) takie systemy wykonują wyraźne odwzorowanie nieliniowe, które jest określone w postaci reguł rozmytych uzyskanych przez przełożenie wiedzy eksperckiej na program komputerowy. Istnieją trzy główne, wyróżniane w dostępnej literaturze typy neuronowo-rozmytych systemów inferencji.

- Kooperacyjny system inferencji rozmytej i sieci neuronowych,
- Równoległy system inferencji rozmytej i sieci neuronowych,
- Hybrydowy system inferencji rozmytej i sieci neuronowych.

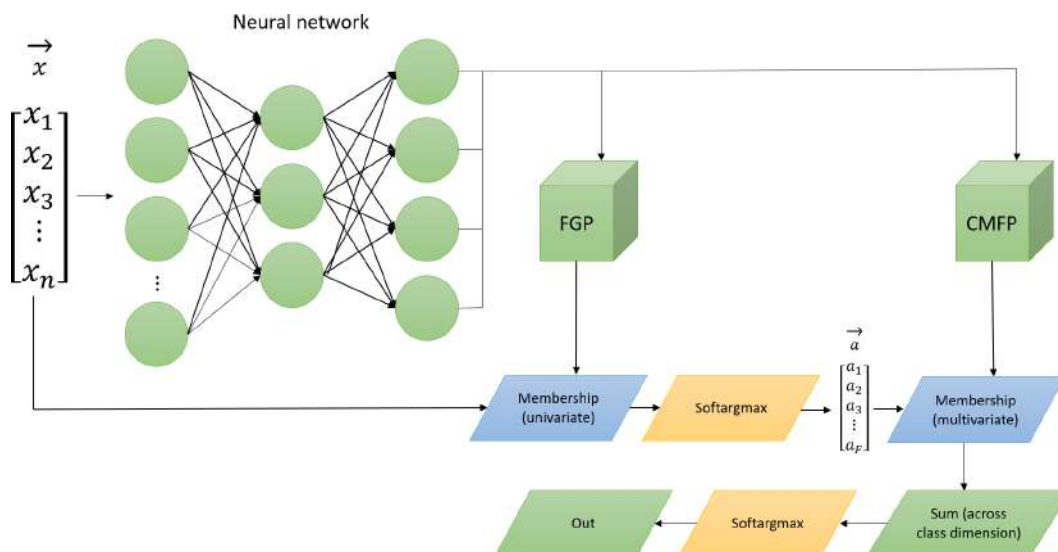
Neuronowo-Rozmyte Systemy Inferencji były kiedyś jedną z najpotężniejszych i najbardziej wymagających technik w uczeniu maszynowym. Niestety, ich granice zostały szybko odkryte i zostały w tyle za bardziej wydajnymi i skalowanymi metodami uczenia maszynowego i głębokiego uczenia. Pierwszym dużym problemem jest fakt, że nadal nie mogą przetwarzać danych nieustrukturyzowanych, takich jak tekst i obrazy. Innym powodem jest ich architektura. Poprzez połączenie wielowarstwowej sieci neuronowej z systemem logiki rozmytej w pojedynczą strukturę, badacze nałożyli ograniczenie dotyczące głębokości sieci neuronowej. Z ostatnich wydarzeń w dziedzinie głębokiego uczenia wiadomo, że te algorytmy, biorąc pod uwagę wystarczającą ilość danych, tworzą poziomy abstrakcji, które tworzą liniową separację danych wejściowych. Ograniczając liczbę warstw sieci neuronowej, zasadniczo ograniczamy jej zdolności reprezentacyjne co sprawia, że taka sieć nie jest w stanie poprawnie oddzielić punktów danych.

3.3 Uczenie reprezentacji rozmytej

Główną ideą tego algorytmu jest to, że głęboka sieć neuronowa oparta na danych wejściowych musi generować kompletny system logiki rozmytej, który jest następnie używany do przewidywania. Nie ma określonych zasad przez ludzi; system konwertuje informacje z danych wejściowych do systemu rozmytego aż po predykcję. Kroki algorytmu są następujące:

1. Ekstrakcja cech,
2. Zdefiniowanie rozmycia cech,
3. Określenie przymiotnikowości cech rozmytych,
4. Konwersja rozmytej przymiotnikowości na dyskretną,
5. Zdefiniowanie rozmycia klas,
6. Określenie przymiotnikowości klas rozmytych,
7. Sumowanie po przynależnościach klas,
8. Obliczenie predykcji.

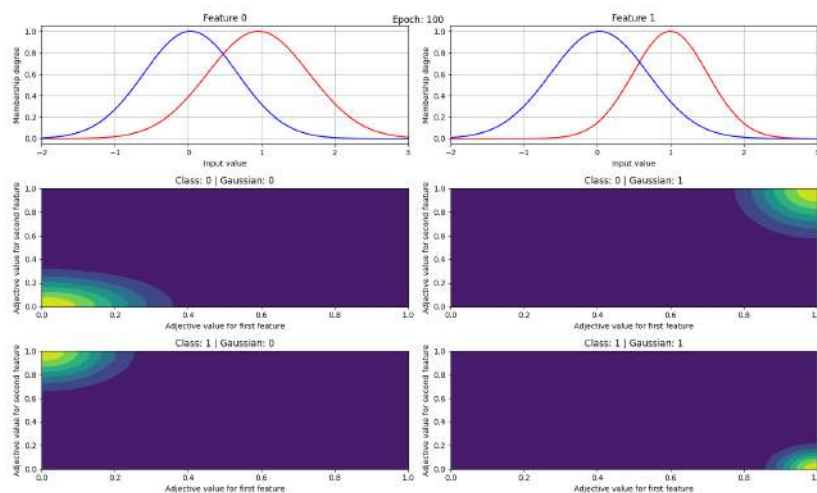
Ogólna architektura systemu została zaprezentowana na rys. 1



Rysunek 1: Architektura systemu.

3.4 Przykład

Jako zweryfikowanie koncepcji, bramka logiczna XOR została zaadaptowana jako problem do rozwiązania. XOR to bramka logiczna, która daje prawdziwe (stan wysoki) wyjście, gdy liczba prawdziwych wejść jest nieparzysta; w przeciwnym razie stan jest fałszywy (stan niski). XOR jest również uważany za najprostszy problem nieliniowy. Aby opisać ten problem przy pomocy przedstawionego algorytmu trzeba najpierw określić ziarnistość cech (liczba przymiotników opisujących cechę), w tym przypadku została ustawiona na dwa (każda cecha może być wysoka lub niska). Stopień ziarnistości przynależności do klasy został również ustawiona na dwa, ponieważ każdą kombinację przymiotników cech można przypisać do dwóch możliwych stanów w ramach jednej klasy (każda klasa może być w stanie wysokim lub niskim). Sieć neuronowa była standardową wielowarstwową siecią neuronową z dwoma ukrytymi warstwami, z których każda składa się ze 128 neuronów, po których następuje funkcja aktywacji ReLU. Model został wytrenowany przy użyciu binarnej funkcji entropii krzyżowej, która jest standardową funkcją straty wykorzystywaną do procesu uczenia głębokich sieci neuronowych do zadania klasyfikacji. Model trenowano przez 100 epok, co dało ostateczną stratę 0,1231. Stan reprezentacji rozmytej w ostatniej epoce przedstawiono na rysunku 2.



Rysunek 2: Reprezentacja rozmyta podczas ostatniej epoki.

4 Conclusions

Niniejsza praca opisuje potencjał zastosowań głębokiego uczenia się w dziedzinie inżynierii biomedycznej. Rozpoczyna się od nakreślenia historii sztucznej inteligencji i rozwoju różnych podejść, takich jak systemy eksperckie, uczenie maszynowe i wreszcie uczenie głębokie. Głębokie uczenie ma ogromny potencjał w dziedzinie inżynierii biomedycznej. Może być używany do zadań takich jak analiza obrazu medycznego, przetwarzanie sygnałów medycznych, odkrywanie leków, klasyfikacja cząsteczek i diagnoza chorób, potencjalnie poprawiając wyniki pacjentów i przyspieszając rozwój nowych technologii medycznych. Jednak ta technologia ma też wiele wad. Uczenie głębokie jest często opisywane jako niewystarczająco przejrzyste czy trudne do zintegrowania z istniejącą wiedzą domenową. Wszystkie te powody są przedmiotem debaty czy aby na pewno, te algorytmy w ich obecnym stanie i ich wykorzystanie w procesach diagnostyczno-terapeutycznych jest etyczne. Z drugiej strony systemy logiki rozmytej, algorytm w stylu systemu eksperckiego, okazały się odporne na próbki spoza dystrybucji, reprezentujących wiedzę w przejrzysty sposób i dających się łatwo modyfikować. Z tego powodu zaproponowano nowe podejście, zwane „uczeniem reprezentacji rozmytej”, które próbuje połączyć głębokie sieci neuronowe i system logiki rozmytej, które można bezpiecznie wdrożyć w zastosowaniach medycznych.

Literatura

- [BMR⁺20] Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. 2020.
- [BZK22] Lucas Beyer, Xiaohua Zhai, and Alexander Kolesnikov. Big vision. Available online at: https://github.com/google-research/big_vision, 2022. Accessed on: .
- [CDW⁺22] Zhe Chen, Yuchen Duan, Wenhai Wang, Junjun He, Tong Lu, Jifeng Dai, and Yu Qiao. Vision transformer adapter for dense predictions. *arXiv preprint arXiv:2205.08534*, 2022.
- [CND⁺22] Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam M. Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Benton C. Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayanan Pillai, Marie Pellat, Aitor Lewkowycz, Erica Oliveira Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathleen S. Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. Palm: Scaling language modeling with pathways. 2022.
- [DCLT18] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.
- [Fan19] Claudio Fanconi. Smiles toxicity. <https://www.kaggle.com/fanconic/smiles-toxicity>, 8 2019.
- [HK19] Michael Haenlein and Andreas Kaplan. A brief history of artificial intelligence: On the past, present, and future of artificial intelligence. *California Management Review*, 61:000812561986492, 07 2019.
- [LBH15] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, May 2015.
- [LLH⁺20] Pengyong Li, Yuquan Li, Chang-Yu Hsieh, Shengyu Zhang, Xianggen Liu, Huanxiang Liu, Sen Song, and Xiaojun Yao. TrimNet: learning molecular representation from triplet messages for biomedicine. *Briefings in Bioinformatics*, 11 2020. bbaa266.
- [Mar18] Gary Marcus. Deep learning: A critical appraisal. *CoRR*, abs/1801.00631, 2018.
- [OEB⁺19] Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. fairseq: A fast, extensible toolkit for sequence modeling. In *Proceedings of NAACL-HLT 2019: Demonstrations*, 2019.
- [PT85] Jiapu Pan and Willis J. Tompkins. A real-time qrs detection algorithm. *IEEE Transactions on Biomedical Engineering*, BME-32(3):230–236, 1985.
- [Rob51] Herbert E. Robbins. A stochastic approximation method. *Annals of Mathematical Statistics*, 22:400–407, 1951.

- [WDC⁺22] Wenhai Wang, Jifeng Dai, Zhe Chen, Zhenhang Huang, Zhiqi Li, Xizhou Zhu, Xiaowei Hu, Tong Lu, Lewei Lu, Hongsheng Li, et al. Internimage: Exploring large-scale vision foundation models with deformable convolutions. *arXiv preprint arXiv:2211.05778*, 2022.
- [YLDG22] Jianwei Yang, Chunyuan Li, Xiyang Dai, and Jianfeng Gao. Focal modulation networks, 2022.
- [YWV⁺22] Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. Coca: Contrastive captioners are image-text foundation models. 2022.
- [YZW22] Haotian Yan, Chuang Zhang, and Ming Wu. Lawin transformer: Improving semantic segmentation transformer with multi-scale representations via large window attention. *arXiv preprint arXiv:2201.01615*, 2022.
- [Zad65] L. A. Zadeh. Fuzzy sets. *Information and Control*, 8(3):338–353, June 1965.
- [ZBE⁺19] Zhen Zhang, Jiajun Bu, Martin Ester, Jianfeng Zhang, Chengwei Yao, Zhi Yu, and Can Wang. Hierarchical graph pooling with structure learning. *arXiv preprint arXiv:1911.05954*, 2019.
- [ZCZX23] Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. Are transformers effective for time series forecasting? 2023.
- [ZSL22] Zhuofan Zong, Guanglu Song, and Yu Liu. Detrs with collaborative hybrid assignments training, 2022.