








Metody i zastosowania sztucznej inteligencji

Mateusz Baran , Joanna Grabska-Chrząstowska ,
Adrian Horzyk , Jaromir Przybyło , Krzysztof Rzecki ,
Tomasz Sośnicki , Ryszard Tadeusiewicz 

AGH Akademia Górniczo-Hutnicza, Wydział Elektrotechniki, Automatyki, Informatyki i Inżynierii Biomedycznej, Kraków

Streszczenie: Na Wydziale Elektrotechniki, Automatyki, Informatyki i Inżynierii Biomedycznej wiele zespołów naukowych rozwija metody w dziedzinie sztucznej inteligencji i jej zastosowań. W prezentowanym tu artykule zostały przedstawione cztery wybrane obszary dotyczące tej problematyki, które są badane i rozwijane w Katedrze Biocybernetyki i Inżynierii Biomedycznej. Pierwszy z tych obszarów to zagadnienia związane ze sztuczną inteligencją wykorzystującą grafowe struktury wiedzy. Jest to unikatowy kierunek badań, przewyższający wiele ograniczeń związanych z użyciem typowych sieci neuronowych. Drugi z obszarów opisuje możliwości, jakie daje programowanie probabilistyczne w sztucznej inteligencji. To także nowe i bardzo obiecujące podejście. Kolejne dwa rozdziały dotyczą przykładów zastosowań. Najpierw omówione zostało konkretne wykorzystanie metod głębokiego uczenia do wykrywania tętna człowieka, a z kolei w ostatnim rozdziale zawarto szeroki przegląd różnych zastosowań metod uczenia nadzorowanego.

Słowa kluczowe: sztuczna inteligencja, uczenie maszynowe, inżynieria biomedyczna

METHODS AND APPLICATIONS OF ARTIFICIAL INTELLIGENCE

Abstract: At the Faculty of Electrical Engineering, Automatics, Computer Science and Biomedical Engineering, many research teams develop methods and applications of artificial intelligence. In the article presented here, four selected problem areas concerning this subject were selected, presented, studied, and developed at the Department of Biocybernetics and Biomedical Engineering. The first of these areas presents the issues of artificial intelligence based on knowledge graph structures. It is a unique research direction that overcomes many limitations related to the use of typical neural networks. The second of the research areas presented here describes the possibilities offered by probabilistic programming in artificial intelligence. It is also a new and very promising approach. The next two chapters are about application examples. First, the specific use of deep learning methods in detecting human heart rate is discussed. In turn, the last chapter provides a broad overview of the various uses of supervised learning.

Keywords: artificial intelligence, machine learning, biomedical engineering

https://doi.org/10.7494/978-83-66727-89-2_5

1. Wstęp

Prezentowany artykuł zawiera niektóre arbitralnie wybrane przykłady prac naukowych prowadzonych w Katedrze Biocybernetyki i Inżynierii Biomedycznej AGH w obszarze metod i zastosowań sztucznej inteligencji. Warto podkreślić, że tego rodzaju prac realizowanych jest w Katedrze znacznie więcej, ale ze względu na objętość tej publikacji wybrano cztery reprezentatywne przykłady, prezentujące kolejno dwa opracowania na temat tworzenia nowych metod sztucznej inteligencji oraz dwa opracowania na temat zastosowań. Nie było możliwe omówienie licznych prac naszego zespołu na temat wykorzystania typowych sieci neuronowych do rozwiązywania problemów z zakresu inżynierii biomedycznej (Grabska-Chrzastowska 2013), a także niektórych prac wykorzystujących metody głębokiego uczenia (Sakr i in. 2022). Mimo to niniejszy artykuł pokazuje główne kierunki badań wszystkich jego autorów, więc celowe jest jego zaprezentowanie na konferencji mającej z założenia pokazać syntetyczny obraz zainteresowań naukowych pracowników Wydziału Elektrotechniki, Automatyki, Informatyki i Inżynierii Biomedycznej.

2. Sztuczna inteligencja wykorzystująca grafowe struktury wiedzy

2.1. Droga do sztucznej inteligencji

Celem informatyki jest reprezentatywne modelowanie wycinków rzeczywistości, co pozwala na automatyczne rozwiązywanie problemów naszej cywilizacji. Powstające na jej gruncie modele mają umożliwić między innymi symulację rzeczywistości oraz predykcję zachowania modelowanych obiektów ze świata rzeczywistego. Modele te mogą służyć również do rozwiązania zadań rozpoznawania, klasyfikacji, detekcji, klasteryzacji, rekomendacji, predykcji i wielu innych. Na gruncie sztucznej inteligencji, mającej na celu modelowanie inteligencji ludzkiej, dążymy więc do budowy modeli symulujących działanie inteligentnych procesów kognitywnych rozgrywających się w naszym umyśle i formujących wiedzę o świecie oraz umożliwiających rozwiązywanie różnych zadań. Wobec tego jednym z fundamentalnych zagadnień przy budowaniu modeli sztucznej inteligencji jest sposób formowania i reprezentacji wiedzy w modelach mających dysponować sztuczną inteligencją. Sposób reprezentacji wiedzy, szybkość dostępu do informacji, jakość generalizacji i skojarzeń wywoływanych na podstawie uformowanej wiedzy decydują o przydatności zbudowanego systemu sztucznej inteligencji. W tym artykule poruszono kilka istotnych kwestii dotyczących reprezentacji danych i łączących je relacji, na podstawie których można uformować wiedzę

i reprezentować ją w specjalistycznych skojarzeniowych strukturach neuronowych, pozwalających na uogólnianie oraz skojarzeniowe wywoływanie powiązanych grup danych. Pokazano również, jak dzięki specyficznej strukturze skojarzeniowej możliwe jest rozwiązywanie wielu zadań sztucznej inteligencji za pomocą tego samego modelu, a nie tylko jednego zadania, jak to ma miejsce w przypadku uczenia nadzorowanego, które buduje model tylko dla ściśle zdefiniowanych danych wejściowych i wyjściowych. Zwrócono uwagę również na adaptacyjność tych modeli oraz łatwość w ich dostosowaniu do nowych danych bez zmuśnego i długotrwałego procesu ponownego ich trenowania. W końcu wysunięto wnioski dotyczące procesu formowania się wiedzy w neuronowych strukturach asocjacyjnych oraz wykorzystania jej do tworzenia systemów sztucznej inteligencji.

2.2. Reprezentacja danych i relacji

W informatyce przyjęło się, że w różnego rodzaju strukturach danych i bazach danych przechowywane i przetwarzane są dane, gdyż są one czymś namacalnym i realnym. Algorytmy operują na danych, ale celem jest zwykle wykrycie istotnych relacji łączących te dane, które prowadzą nas do wartościowych wniosków i wyników, zapisywanych w postaci danych wyjściowych. Warto jednak zwrócić uwagę na to, że dane nabierają sensu oraz umożliwiają formowanie informacji dopiero wtedy, gdy powiązane są pewnymi relacjami, np. liczba 2022 może mieć bardzo różne znaczenie w zależności od kontekstu jej wykorzystania. W naszych biologicznych sieciach neuronowych posługujemy się skojarzeniami, które pozwalają nam takie dane jak np. liczbę 2022 umieścić w pewnym kontekście sytuacji, potrzeb, faktów oraz naszej wiedzy, a wtedy uruchamia się lawina skojarzeń, która może prowadzić do różnych konkluzji. Ze względu na to, że wiedza poszczególnych osób różni się, kombinacje danych wejściowych mogą prowadzić do uruchomienia różnych lawin skojarzeń oraz różnych wniosków z tego płynących u różnych osób. Tak więc komunikacja międzyludzka wymaga zdefiniowania odpowiednio bogatego kontekstu, by mogło nastąpić zrozumienie informacji przekazywanych odbiorcom przez nadawców. Im bardziej wiedza odbiorców i nadawców jest podobna i zbieżna, tym prościej komunikować się, gdyż reprezentowane są podobne dane oraz podobne relacje je łączące.

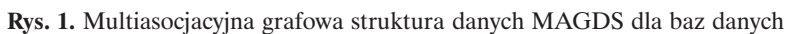
Reprezentacja relacji ma więc fundamentalne znaczenie w procesie formowania wiedzy oraz interpretacji danych do nas docierających. Wydaje się jednak, że współczesna informatyka nie potraktowała jeszcze tej kwestii wystarczająco poważnie, skupiając się na danych, strukturach danych, bazach danych czy hurtowaniach danych. Okazuje się jednak, że klasyczne struktury danych różnią się właśnie sposobem reprezentacji

niektórych relacji, jakie występują pomiędzy danymi, a dzięki temu umożliwiają zmianę sposobu ich przetwarzania lub decydują o szybkości dostępu do nich.

Dla przykładu rozważmy tablicę danych posortowanych i nieposortowanych. W pierwszej z nich występuje niejawnie reprezentacja relacji porządkowania elementów, która umożliwia szybkie wyszukiwanie w czasie logarytmicznym, zaś w tej drugiej brak tej relacji uniemożliwia zastosowanie takiego sposobu wyszukiwania. Podobnie jest w strukturach znacznie bogatszych, np. obecnie coraz bardziej popularnych grafach wiedzy czy różnych modelach pamięci (Horzyk i in. 2017, Basawaraj i in. 2019, Starzyk i in. 2021), które mają możliwość reprezentacji wielu różnych relacji, a dzięki temu kojarzą ze sobą dane na wiele różnych sposobów, co znacząco przyspiesza ich wyszukiwanie oraz wnioskowanie na ich podstawie. Podobnie nasze neuronowe struktury mózgowe są niejednorodnymi, rzadkimi grafami zbudowanymi z różnych typów neuronów, tworzących i rekonfigurujących połączenia pomiędzy sobą przez całe nasze życie, gdyż ma to związek z procesami formowania się naszej wiedzy oraz rozwojem inteligencji. Przeprowadzone do tej pory badania nad reprezentacją dodatkowych relacji w specjalnych strukturach skojarzeniowych (Horzyk i Starzyk 2018, Horzyk i in. 2020) wykazały możliwość przyspieszenia operacji wykonywanych na danych, zaś brak reprezentacji relacji zmusza informatyków do przygotowania specjalnych procedur pozwalających na odnalezienie w danych pożądaných relacji, aby możliwe było ich wykorzystanie. Im więcej relacji jest dostępnych bezpośrednio w strukturze, tym mniej operacji wyszukiwania musimy wykonywać i tym szybciej procesy wnioskowania mogą zachodzić. Dzięki temu, że w naszych biologicznych sieciach neuronowych relacje reprezentowane są bardzo bogato, nasze bardzo wolno działające neurony (w porównaniu do szybkości współczesnych procesorów) są w stanie reagować nawet na bardzo złożone sytuacje błyskawicznie.

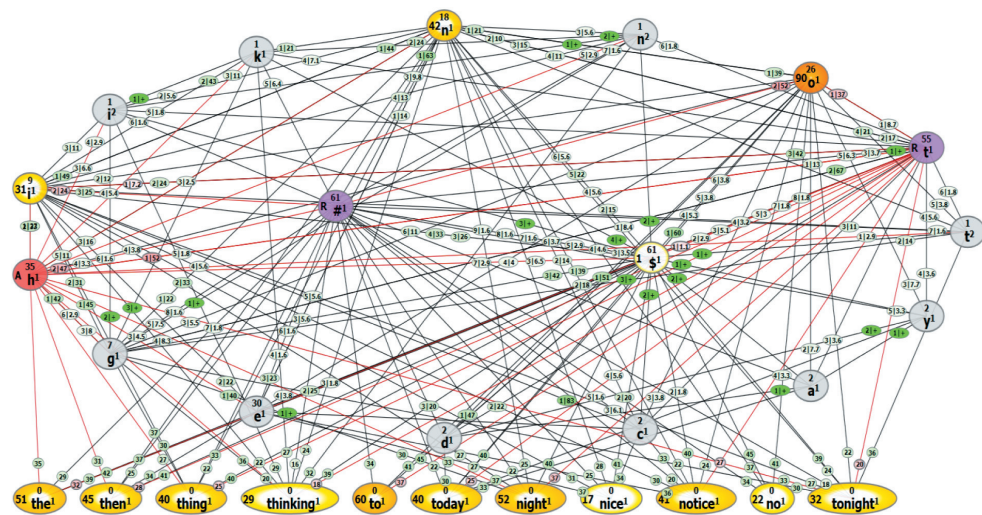
2.3. Modelowanie wiedzy

Modelowanie wiedzy wymaga wykorzystania takich struktur, które będą zdolne nie tylko reprezentować dane, ale również bogactwo łączących je relacji, by mogły one być błyskawicznie użyte w procesie wnioskowania, a także automatycznie prowadzić do wniosków bazujących na nich. Formowanie struktur wiedzy wymusza więc określenie podstawowych relacji, które powinny być w nich odwzorowane, jak np. podobieństwo, które umożliwia uogólnianie wiedzy na podobne obiekty. Istotne są również relacje definiujące bardziej złożone obiekty na podstawie prostszych czy też te wskazujące różne specyficzne powiązania pomiędzy obiektami oraz zależności czasowo-przestrzenne. Przykład takiej struktury wygenerowanej dla bazy danych gromadzącej dane i relacje pomiędzy pacjentami, pielęgniarkami, lekarzami oraz pokojami szpitalnymi, wraz z określeniem wybranych atrybutów, pokazany został na rysunku 1.



W strukturze tej określono imiona i nazwiska lekarzy, pielęgniarek i pacjentów oraz zagregowano ich reprezentację niezależnie od tabeli bazodanowej, w której występowały. Podobnie postąpiono z danymi określającymi staż pracy lekarzy i pielęgniarek w latach. Dzięki agregacji wartości atrybutów z różnych tabel oraz ich reprezentacji za pomocą jednej kategorii danych w strukturze asocjacyjnej automatycznie uzyskujemy możliwość formowania różnych ciekawych wniosków opierających się na posiadaniu tego samego imienia, nazwiska czy też podobnego stażu pracy z uwzględnieniem

innych atrybutów lub bez nich. Dla przykładu, jeśli obchodzimy imieniny, to z łatwością wywołamy w takim grafie skojarzenia dla wszystkich osób niezależnie od tego czy są lekarzami, pielęgniarkami czy pacjentami. Jeśli zaś poszukujemy lekarzy o odpowiednim stażu, wtedy uwzględnienie odpowiednio szerszego kontekstu stymulacji węzłów w takim grafie sprawi, że nie będą brane pod uwagę pielęgniarki, ale tylko lekarze. Istotną zaletą w stosunku do filtrowania w relacyjnych bazach danych jest możliwość znalezienia obiektów, które nie tylko dokładnie spełniają kryteria wyszukiwania, ale są do nich w znacznym stopniu podobne. Kolejnym krokiem jest zdefiniowanie reaktywności elementów takich struktur, czyli zdolności do automatycznej odpowiedzi i pobudzania innych skojarzonych węzłów w przypadku osiągnięcia sumarycznej stymulacji przekraczającej pewien próg ich aktywacji. Wtedy takie węzły stają się skojarzeniowymi neuronami impulsowymi. Przykład skojarzeniowego grafu neuronowego (Horzyk i in. 2017, Horzyk i Starzyk 2018) operującego na słowach języka naturalnego widoczny jest na rysunku 2, na którym wskazano stany reaktywnych neuronów skojarzeniowych podlegających procesom ładowania, relaksacji i refrakcji w czasie.



Rys. 2. Grafowa, multiasocjacyjna sieć neuronowa dla sekwencji

W strukturach grafowych jesteśmy w stanie reprezentować zagregowane dane, eliminując duplikaty, zliczając je i na ich podstawie określając skojarzeniową siłę oddziaływania poszczególnych węzłów na siebie. Zagregowane reprezentacje danych mogą zostać wykorzystane do definiowania obiektów, które stają się automatycznie skojarzone na skutek ich zdefiniowania za pomocą węzłów równocześnie definiujących też inne obiekty. Możemy określić siłę skojarzenia różnych obiektów dzięki takim połączeniom ważonym w zależności od podobieństwa lub liczby reprezentowanych obiektów. Sty-

mulowane neurony w takich grafach uruchamiają setki skojarzeń na bazie stymulacji pochodzących od połączonych neuronów, przy czym te stymulacje rozgrywają się w czasie z różną częstotliwością, co umożliwia dyskryminację różnych danych wejściowych oraz rozwiązywanie zadań rozpoznawania, klasyfikacji, klasteryzacji, rekomendacji i wielu innych z wykorzystaniem tego samego modelu. Dodatkowym atutem tego rodzaju grafów neuronowych jest więc możliwość ich budowania bez określenia z góry zadania, do którego będą stosowane, jak jest to wymagane w przypadku uczenia nadzorowanego. Tutaj cel możemy określić później, a graf zawsze można zaktualizować i rozbudować o nowe dane lub usunąć z niego te nieaktualne. Tworzenie takich grafów na skutek procesów agregacji i zliczania duplikatów oraz łączenia podobnych danych automatycznie zwiększa liczbę reprezentowanych w grafie relacji w stosunku do tradycyjnych relacyjnych baz danych, a to ułatwia, przyspiesza i automatyzuje wnioskowanie.

2.4. Perspektywy rozwoju sztucznej inteligencji wykorzystującej grafowe struktury wiedzy

Sztuczna inteligencja może w przyszłości formować się w automatycznie tworzonych reaktywnych, skojarzeniowych strukturach neuronowych, które dzięki możliwości reprezentacji wiedzy o problemach na skutek odpowiedniej reprezentacji danych i relacji umożliwią inteligentne wnioskowanie dla dowolnego kontekstu wejściowego. Istotą tych procesów jest czas, w którym procesy zachodzą w tych strukturach grafowych, gdyż wyniki zależne są od krotności i czasu aktywacji neuronów (Horzyk i Starzyk 2018). Warto nadmienić, że nasze mięśnie też napinają się w zależności od intensywności ich stymulacji przez system nerwowy, więc taka interpretacja wyników na podstawie liczby stymulacji jest biologicznie uzasadniona.

Sztuczna inteligencja, aby się rozwijać, potrzebuje odpowiedniej reprezentacji wiedzy, którą będzie mogła wykorzystywać w swoich procesach wnioskowania oraz generowania wyników. Wiedza zaś na podobieństwo struktur biologicznych może formować się w adaptacyjnych strukturach neuronowych budowanych z neuronów nowych generacji, które odpowiednio modelują istotne procesy biologiczne, nie tylko umożliwiając adaptację wag, ale również dostosowując strukturę sieci. Działają one w czasie i umożliwiają modelowanie skojarzeń i ich wywoływanie. Automatyczny rozwój sztucznej inteligencji wymaga oczywiście również innych czynników nieomówionych w tym artykule, np. motywujących ją do poszerzania wiedzy w konkretnym kierunku, lecz te tematy zasługują na przedstawienie w oddzielnym artykule.

Obecnie najczęściej stosowane uczenie nadzorowane sieci neuronowych umożliwia często bardzo efektywne rozwiązywanie specyficznych problemów nawet na poziomie ludzkim, jednak raz nauczona sieć nie nadaje się bez przetrenowania do innych zadań. Nie ma więc zdolności automatycznego rozwiązywania różnych problemów na

podstawie uformowanej wiedzy w procesie uczenia, co jest istotnym ograniczeniem tego podejścia, które blokuje rozwój sztucznej inteligencji, sprowadzając ją do inteligencji obliczeniowej. Niemniej współcześnie korzyści wynikające z jej stosowania w wielu dziedzinach życia są znaczące, więc warto dalej ją rozwijać i wykorzystywać do rozwiązywania rzeczywistych problemów, niezależnie od dalszego rozwoju sztucznej inteligencji wykorzystującej skojarzeniowe modele wiedzy.

3. Programowanie probabilistyczne w sztucznej inteligencji

W ostatnich latach nastąpił znaczący rozwój technik wnioskowania w probabilistycznych sieciach grafowych oraz algorytmów programowania różniczkowalnego. Osiągnięcia te znacząco uprościły i upowszechniły stosowanie zaawansowanych metod bayesowskich w różnych dziedzinach nauki (McElreath 2020). Możliwe stało się tworzenie skomplikowanych, ale w znacznym stopniu interpretowalnych modeli i ich efektywne dopasowanie do danych z wykorzystaniem szerokiego zakresu metod typu *Markov chain Monte Carlo* (MCMC) – począwszy od klasycznych wariantów algorytmu Metropolisa–Hastingsa, aż po np. geometryczne warianty metody *Hamiltonian Monte Carlo*.

Podstawowym elementem w programowaniu probabilistycznym jest model opisujący, w jaki sposób znane bądź nieznanne parametry przekładają się na obserwacje. Modele te mogą mieć charakter bardziej deklaratywny, jak w przypadku technik ściślej związanych z probabilistycznymi modelami grafowymi, lub też mieć formę proceduralnego programu komputerowego. W tym drugim wariacie modelem jest program, który wygląda podobnie jak w klasycznych językach proceduralnych, dzięki czemu jest względnie prosty do zrozumienia. Zastosowanie odpowiednich technik transformacji kodu i śledzenia wykonania pozwala na próbkowanie rozkładów *a posteriori* parametrów modelu. Jest to proces, dzięki któremu zbierane dane przekładane są na informacje, które można potem wykorzystać do różnego rodzaju wnioskowania probabilistycznego.

Pewną wadą zaawansowanych metod bayesowskich jest podatność metod MCMC na dążenie do nieprawdziwego rozkładu *a posteriori*. Problem ten jest jednak minimalizowany dzięki dostępności coraz liczniejszych technik analizy, które pozwalają na identyfikację problemów, oraz coraz bardziej zaawansowanym algorytmom potrafiącym radzić sobie ze skomplikowanymi rozkładami.

Programowanie probabilistyczne znalazło liczne zastosowania w dziedzinie sztucznej inteligencji. Jednym z przykładów jest optymalizacja hiperparametrów (Neiswanger i in. 2019). Znany problemem występującym między innymi w uczeniu nadzorowanym jest konieczność wyboru np. architektury sieci neuronowej czy parametrów algorytmu optymalizacji funkcji straty. Jest to problem, w którym mamy określoną metafunkcję straty opisującą, na ile dobry jest konkretny zestaw hiperparametrów. Optymalizacja

staje się bardziej skomplikowana, ponieważ wartości metafunkcji straty możemy tylko estymować (gdyż mamy ograniczone środki na trening modelu), a proces ten jest bardzo czasochłonny. Metody bayesowskie, dzięki precyzyjnemu modelowaniu rozkładów prawdopodobieństwa, mogą potencjalnie bardziej efektywnie wykorzystywać wyniki ewaluacji hiperparametrów niż np. algorytmy ewolucyjne (Bischl i in. 2021).

Innym przykładem wykorzystania programowania probabilistycznego jest tworzenie agentów w procesach decyzyjnych Markowa czy uczeniu ze wzmocnieniem. Pozwala ono na efektywne przewidywanie nieobserwowanych elementów stanu bądź środowiska, w którym funkcjonuje agent. Rozwijane są obecnie systemy mające pozwolić na praktyczne zastosowanie takich rozwiązań, w tym do rozwiązywania problemu jednoczesnej lokalizacji i mapowania, gdzie wielomodowe rozkłady prawdopodobieństwa stanowią duże wyzwanie dla bardziej klasycznych metod. Ponadto konieczne jest uwzględnienie w metodach probabilistycznych geometrii grup Liego służących do opisu położenia i orientacji obiektów w przestrzeni. Przydatne mogą tu być stale rozwijane narzędzia do obliczeń statystycznych w geometrii różniczkowej (Axen i in. 2021).

Programowanie probabilistyczne ma też liczne zastosowania biomedyczne, między innymi w analizie danych eksperymentalnych, planowaniu leczenia czy wykrywaniu nadużyć. Potencjalnie może też być wykorzystywane w kontroli aparatury medycznej czy analizie sygnałów biometrycznych (Rzecki i Baran 2021).

Podsumowując, należy podkreślić, że programowanie probabilistyczne znacząco rozszerza zakres modeli probabilistycznych, jakie są dostępne dla przeciętnego naukowca zajmującego się sztuczną inteligencją. Dzięki rozbiciu problemu modelowania na niezależne elementy (między innymi tworzenie modelu, estymacja parametrów, liczenie gradientu, optymalizacja, akwizycja danych, wnioskowanie na podstawie modelu) możliwe jest wykorzystanie najnowszych technik z poszczególnych obszarów do jak najlepszego rozwiązania sformułowanego problemu. Szybki rozwój narzędzi i coraz większa ich przystępność sprawia, że jest to temat wart zainteresowania.

4. Wykorzystanie metod głębokiego uczenia do wykrywania tętna człowieka

4.1. Wstęp

Wideopletyzmografia (VPG) stała się ostatnio popularna jako metoda bezkontaktowego pomiaru biosygnałów, w tym parametrów sercowo-naczyniowych (Kranjec i in. 2014). Technika ta – w skrócie – umożliwia zdalny pomiar tętna (HR) lub częstości oddechów (BR) użytkownika za pomocą zwykłej kamery, która obserwuje twarz lub inne obszary skóry osoby badanej. Sygnał VPG jest tworzony przez uśrednienie wartości pikseli wewnątrz wybranego obszaru zainteresowania (ROI) dla kolejnych klatek

wideo. Zaletą takiego podejścia w porównaniu ze standardową techniką PPG jest to, że nie wymaga niewygodnych akcesoriów do noszenia i umożliwia łatwe dostosowanie do różnych wymagań w różnych zastosowaniach.

W rzeczywistych zastosowaniach wiarygodne oszacowanie tętna (HR) na podstawie wideo jest skomplikowanym zadaniem, ponieważ wiele czynników może zanieczyścić sygnał VPG artefaktami. Na wartości pikseli wokół obszaru twarzy mogą mieć wpływ ruchy głowy (pochylenie, zmiana pozycji) oraz mimika twarzy, co skutkuje artefaktami sygnału. Zmiany w otoczeniu, takie jak fluktuacje oświetlenia (światła w pomieszczeniach, migotanie ekranów komputerów) i wewnętrzne szумы aparatu cyfrowego, mogą również wpływać na dokładność szacowania pulsu. Dlatego też algorytm wyznaczania tętna za pomocą metody VPG jest wieloetapowy i składa się z następujących kroków:

1. Wybór ROI: wybór i śledzenie dla każdej klatki wideo regionu zainteresowania (ROI) zawierającego informacje związane z tętnem (najczęściej obszar skóry człowieka).
2. Formowanie sygnału $y_0(n)$ ze składowych koloru obrazu: uśrednienie wartości pikseli wewnątrz ROI dla kolejnych ramek wideo.
3. Wstępne przetwarzanie sygnału, np. usuwanie trendu sygnału.
4. Ekstrakcja sygnału VPG: na podstawie $y_0(n)$ przeprowadzane jest formowanie sygnału VPG(n) jako kombinacja poszczególnych składowych kolorów, np. G (składowa zielona), analiza składowych niezależnych (ICA) (Poh i in. 2011) lub metoda POS (Wang i in. 2016).
5. Postprocessing sygnału VPG, tj. filtrowanie pasmowo-przepustowe.
6. Estymacja częstości tętna: metody czasowe lub częstotliwościowe.

4.2. Zastosowanie metod z dziedziny sztucznej inteligencji

W pracach prowadzonych w Katedrze Biocybernetyki i Inżynierii Biomedycznej AGH zaproponowano kilka usprawnień algorytmu wykrywania tętna, w tym opartych na sztucznej inteligencji (głębokie sieci neuronowe).

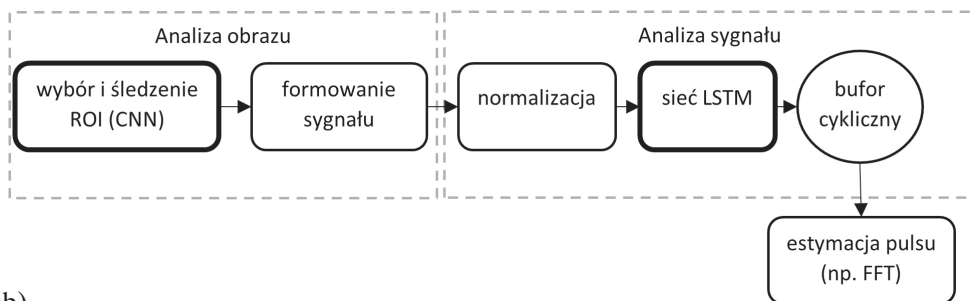
W swojej pracy Przybyło (2019) zaproponował system przeznaczony do ciągłego monitorowania tętna użytkownika podczas typowych scenariuszy interakcji człowiek–komputer (tj. pracy przy komputerze). Przedstawiono nową metodę ekstrakcji sygnału opartą na reprezentacji obrazu wzmacniającej składową zieloną (ExG). Wyniki pokazują, że metoda ExG zapewnia akceptowalną dokładność, a jednocześnie jest znacznie szybsza w obliczeniach niż inne najnowocześniejsze metody (tj. ICA).

W kolejnym artykule Przybyło (2022) przedstawia metodę pomiaru tętna z wykorzystaniem głębokich sieci neuronowych. Schemat blokowy działania algorytmu przedstawiono na rysunku 3. Sztuczna inteligencja została wykorzystana na dwóch etapach algorytmu wyznaczania tętna.

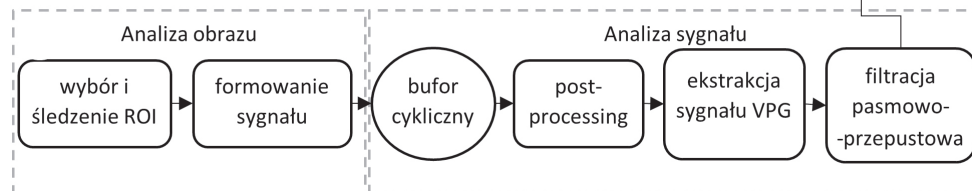
Pierwszy z nich to wybór ROI, który przeprowadzany jest za pomocą konwolucyjnej sieci głębokiej (CNN) o architekturze MobileNetV2 pozwalającej na wykrywanie obszaru twarzy na obrazie.

Na kolejnym etapie wykorzystano sieć głęboką typu LSTM (*long short-term memory*). Sieć LSTM jest rodzajem architektury rekurencyjnej sieci neuronowej (RNN), która jest w stanie nauczyć się długoterminowej zależności między krokami czasowymi w szeregu czasowych a danymi sekwencyjnymi. Ma ona na celu zastąpienie etapów wstępnego i końcowego przetwarzania sygnału (usuwanie trendu, formowanie sygnału, filtracja pasmowo-przepustowa), a w konsekwencji zmniejszenie liczby parametrów algorytmu, które wpływają na dokładność estymacji tętna.

a)



b)



Rys. 3. Schemat blokowy algorytmu: a) proponowane rozwiązanie oparte na sieciach głębokich; b) klasyczne rozwiązania estymacji pulsu. Pogrubioną kreską zaznaczono etapy algorytmu wykorzystujące głębokie sieci neuronowe

Zaproponowana metoda przewyższa algorytm oparty na analizie składowej zielonej (G) i zapewnia wyniki porównywalne z najnowocześniejszymi metodami, takimi jak analiza składowych niezależnych ICA i metoda POS.

5. Przykłady zastosowania uczenia nadzorowanego

Uczenie nadzorowane (*supervised learning*) (Aalst 2014) nazywane jest inaczej uczeniem z nauczycielem. Polega na budowie modelu, który możliwie dobrze przewi-
dzi poprawny wynik, przy czym poprawność wyników określana jest przez nadzorcę

zwanego nauczycielem. Modelem nazywamy metodę sztucznej inteligencji wraz z ustalonymi parametrami. Metoda uczenia maszynowego opisywana jest hiperparametrami, czyli wartościami ustalonymi odgórnie, oraz parametrami, czyli zmiennymi wewnętrznymi metody. Algorytm uczenia dostosowuje parametry budowanego modelu na podstawie analizy wyników oczekiwanych na wyjściu, dla podanych na wejście modelu danych objaśniających. Uczenie nadzorowane wymaga, aby zbiór danych do nauki zawierał zarówno zmienne objaśniające, jak i zmienne objaśniane. Celem algorytmu jest takie ustawienie modelu, aby zwracane przez niego wyniki dla danych ze zbioru testowego jak najbardziej odzwierciedlały wyniki uzyskiwane w trakcie treningu.

Wynikiem działania modelu może być przypisanie etykiety do danych, czyli odpowiedź o charakterze jakościowym (*qualitative*) (James i in. 2013). Zmienne typu jakościowego przyjmują wartości z konkretnego zbioru. Przykładem tutaj może być przynależność do kategorii – odpowiedź w ankiecie typu „bardzo zły / średni / bardzo dobry”. Innym rezultatem działania modelu może być odpowiedź w postaci liczbowej. Zmienna liczbową jest zmienną typu ilościowego (*quantitative*) (James i in. 2013). Zmienne ilościowe opisują wielkości mierzalne, jak np. wiek, wagę, długość czy prawdopodobieństwo. Problemy, w których na wyjściu powinna pojawić się odpowiedź ilościowa, są zazwyczaj problemami regresyjnymi, a problemy zwracające odpowiedzi w postaci jakościowej są problemami klasyfikacyjnymi.

Zazwyczaj wybiera się metodę uczenia maszynowego na podstawie tego, czy oczekiwana odpowiedź jest ilościowa, czy jakościowa, np. można zastosować regresję liniową, gdy oczekuje się na wyjściu odpowiedzi ilościowej, lub regresję logistyczną, gdy oczekuje się na wyjściu odpowiedzi jakościowej. Wybór typu zmiennych objaśniających jest mniej ważny. Większość metod uczenia można wykorzystać niezależnie od typu zmiennej objaśniającej pod warunkiem zastosowania odpowiedniej metody przetwarzania wstępnego. Przykładowo zmienne jakościowe mogą być odpowiednio kodowane przed przeprowadzeniem analizy.

Klasyfikacja (*classification*) (Han i in. 2012, Mohri i in. 2012) jest formą analizy danych, w której modele przyporządkowują dyskretne kategorie do danych w postaci zestawów cech. Modele nazywane są klasyfikatorami, a kategorie klasami. Większość klasycznych metod klasyfikacji charakteryzuje się niewielką liczbą hiperparametrów. Przykładami mogą być drzewa decyzyjne (DT), lasy losowe (RF), metoda k -najbliższych sąsiadów (kNN), naiwny klasyfikator Bayesa, probabilistyczne sieci neuronowe (PNN), maszyna wektorów nośnych (SVM).

Regresja (*regression*) (Han i in. 2012) jest formą analizy danych mającą na celu wyznaczenie sposobu estymowania zmiennej objaśnianej jako zmiennej ilościowej na podstawie zmiennych objaśniających. Najczęściej przyjmuje ona wartości ciągłe. W przeciwieństwie do klasyfikacji, wyznaczającej znane klasy, regresja może służyć do

przewidywania brakujących lub niedostępnych wartości danych liczbowych na podstawie zebranych danych. Proste modele regresji, jak np. model regresji liniowej, mogą być definiowane przez jeden lub dwa parametry, ale istnieją skomplikowane modele regresji wykorzystujące uczenie głębokie, które mogą wymagać milionów parametrów.

Klasyfikacja i regresja są podobnymi problemami i ta sama idea może być wykorzystana w celu konstrukcji algorytmów regresji i klasyfikacji. Przykładowo odpowiednikiem maszyny wektorów nośnych SVM jest SVR (*support vector regression*), odpowiednikami lasów losowych są lasy regresji, sieci neuronowe mogą na wyjściu zwracać zmienne ilościowe, wykonując regresję, lub ich wyjście może być przekształcone do jakościowego i w ten sposób realizować klasyfikację.

Uczenie nadzorowane ma wiele zastosowań. Przykładowymi dziedzinami mogą być wymienione poniżej:

- Biometria – dziedzina, w której zbierane są dane o ludzkim ciele, takie jak wygląd tęczy, linii papilarnych, twarzy, płatka ucha, a także sposób zachowania człowieka, jak wykonywanie gestów palcem na ekranie dotykowym (Rzecki i in. 2017) czy dłonią w rękawicy (Pławiak i in. 2016). Dane te wykorzystywane są w celu wytrenowania systemu do rozpoznawania użytkowników, np. na potrzeby uwierzytelniania w systemie operacyjnym telefonu czy banku.
- Rozpoznawanie materiałów za pomocą laserowo indukowanej spektroskopii emisyjnej (LIBS – *laser induced breakdown spectroscopy*) (Rzecki i in. 2018) – uczenie maszynowe, w którym dane w postaci zarejestrowanego widma światła powstałego po uderzeniu laserem wykorzystywane są do wytrenowania systemu rozpoznawania materiałów.
- Rozpoznawanie mowy – do tej dziedziny należą aplikacje uczące się brzmienia głosu, reagujące na wypowiedziane imię czy słowo kluczowe.
- Radiologia – dziedzina, w której tomografia komputerowa, rezonans magnetyczny, zdjęcie rentgenowskie wykorzystywane są w celach wykrywania nieprawidłowości spowodowanych np. chorobami lub zwyrodnieniami.
- Detekcja obiektów – systemy uczące się wyglądu poszczególnych klas, jak konkretne kształty, zwierzęta, przedmioty. Wytrenowany system może wyszukiwać elementy należące do klas na innych obrazach. Dobrym przykładem jest tutaj wyszukiwarka Google, która potrafi klasyfikować zdjęcia – do tworzenia zbioru treningowego opisanych danych wykorzystuje dane z mechanizmu *captcha*.
- Cyberbezpieczeństwo – systemy bezpieczeństwa (IPS – *intrusion prevention system*) i analizy ruchu (NTA – *network traffic analysis*) uczące się rozpoznawania ataków czy kierowania ruchem na podstawie zebranych i oznaczonych danych. Systemy pocztowe do wykrywania spamu również są inteligentnymi rozwiązaniami, które uczą się wraz z analizą nowych wiadomości.

Podsumowując, w uczeniu nadzorowanym celem jest jak najlepsze przewidywanie odpowiedzi dla nowych danych lub lepsze zrozumienie relacji pomiędzy zmiennymi objaśniającymi a zmiennymi objaśnianymi.

Literatura

- Aalst W.M.P., van der, 2014, *Process Mining: Discovery, Conformance and Enhancement of Business Processes*, Springer, Berlin. <https://doi.org/10.1007/978-3-642-19345-3>.
- Axen S.D., Baran M., Bergmann R., Rzecki K., 2021, *Manifolds.jl: an extensible Julia framework for data analysis on Manifolds*. <https://arxiv.org/abs/2106.08777> [dostęp: 23.03.2022].
- Basawaraj, Starzyk J.A., Horzyk A., 2019, *Episodic memory in minicolumn associative knowledge graphs*, IEEE Transactions on Neural Networks and Learning Systems, vol. 30(11), s. 3505–3516. <https://doi.org/10.1109/TNNLS.2019.2927106>.
- Bischi B., Binder M., Lang M., Pielok T., Richter J., Coors S., Thomas J., Ullmann T., Becker M., Boulesteix A.L., Deng D., Lindauer M., 2021, *Hyperparameter optimization: foundations, algorithms, best practices and open challenges*. <https://arxiv.org/abs/2107.05847> [dostęp: 23.03.2022].
- Grabska-Chrząstowska J., 2013, *Neural network approach to incomplete data applied to assessing cardiac health*, Computing in Cardiology, vol. 40, s. 499–502.
- Han J., Kamber M., Pei J., 2012, *Data Mining: Concepts and Techniques*, 3rd ed., Elsevier – Morgan Kaufmann, Amsterdam – Boston.
- Horzyk A., Starzyk J.A., 2018, *Multi-class and multi-label classification using associative pulsing neural networks*, [w:] *2018 International Joint Conference on Neural Networks (IJCNN)*, IEEE, s. 427–434. <https://doi.org/10.1109/IJCNN.2018.8489176>.
- Horzyk A., Starzyk J.A., Graham J., 2017, *Integration of semantic and episodic memories*, IEEE Transactions on Neural Networks and Learning Systems, vol. 28(12), s. 3084–3095. <https://doi.org/10.1109/TNNLS.2017.2728203>.
- Horzyk A., Bulanda D., Starzyk J.A., 2020, *ASA-graphs for efficient data representation and processing*, International Journal of Applied Mathematics and Computer Science (AMCS), vol. 30(4), s. 717–731. <https://doi.org/10.34768/amcs-2020-0053>.
- James G., Witten D., Hastie T., Tibshirani R., 2013, *An Introduction to Statistical Learning: with Applications in R*, Springer Texts in Statistics, vol. 103, Springer, New York. <https://doi.org/10.1007/978-1-4614-7138-7>.
- Kranjec J., Beguš S., Geršak G., Drnovšek J., 2014, *Non-contact heart rate and heart rate variability measurements: A review*, Biomedical Signal Processing and Control, vol. 13, s. 102–112. <https://doi.org/10.1016/j.bspc.2014.03.004>.
- McElreath R., 2020, *Statistical Rethinking: A Bayesian Course with Examples in R and STAN*, CRC Press – Francis & Taylor, Boca Raton – Oxon.

- Mohri M., Rostamizadeh A., Talwalkar A., 2012, *Foundations of Machine Learning*, Adaptive Computation and Machine Learning, The MIT Press, Cambridge.
- Neiswanger W., Kandasamy K., Poczos B., Schneider J., Xing E., 2019, *ProBO: versatile Bayesian optimization using any probabilistic programming language*. <https://arxiv.org/abs/1901.11515> [dostęp: 23.03.2022].
- Pławiak P., Sośnicki T., Niedźwiecki M., Tabor Z., Rzecki K., 2016, *Hand body language gesture recognition based on signals from specialized glove and machine learning algorithms*, IEEE Transactions on Industrial Informatics, vol. 12(3), s. 1104–1113. <https://doi.org/10.1109/TII.2016.2550528>.
- Poh M.Z., McDuff D.J., Picard R.W., 2011, *Advancements in noncontact, multiparameter physiological measurements using a webcam*, IEEE Transactions on Biomedical Engineering, vol. 58(1), s. 7–11. <https://doi.org/10.1109/TBME.2010.2086456>.
- Przybyło J., 2019, *Continuous distant measurement of the user's heart rate in human-computer interaction applications*, Sensors, vol. 19(19), 4205. <https://doi.org/10.3390/s19194205>
- Przybyło J., 2022, *A deep learning approach for remote heart rate estimation*, Biomedical Signal Processing and Control, vol. 74, 103457. <https://doi.org/10.1016/j.bspc.2021.103457>.
- Rzecki K., Baran M., 2021, *Application of elastic shape analysis to user authentication and identification*, IEEE Transactions on Emerging Topics in Computing, vol. 10, iss. 2, s. 1157–1165. <https://doi.org/10.1109/TETC.2021.3074242>.
- Rzecki K., Pławiak P., Niedźwiecki M., Sośnicki T., Leśkow J., Ciesielski M., 2017, *Person recognition based on touch screen gestures using computational intelligence methods*, Information Sciences, vol. 415–416, s. 70–84. <https://doi.org/10.1016/j.ins.2017.05.041>.
- Rzecki K., Sośnicki T., Baran M., Niedźwiecki M., Król M., Łojewski T., Acharya U.R., Yildirim Ö., Pławiak P., 2018, *Application of computational intelligence methods for the automated identification of paper-ink samples based on LIBS*, Sensors, vol. 18(11), 3670. <https://doi.org/10.3390/s18113670>.
- Sakr A.S., Pławiak P., Tadeusiewicz R., Hammad M., 2022, *Cancelable ECG biometric based on combination of deep transfer learning with DNA and amino acid approaches for human authentication*, Information Science, vol. 585, s. 127–143. <https://doi.org/10.1016/j.ins.2021.11.066>.
- Starzyk J.A., Jaszuk M., Maciura Ł., Horzyk A., 2021, *Concurrent associative memories with synaptic delays*, IEEE Transactions on Neural Networks and Learning Systems, vol. 32(8), s. 3736–3747. <https://doi.org/10.1109/TNNLS.2020.3041048>.
- Wang W., Brinker A.C., den, Stuijk S., de Haan G., 2016, *Algorithmic principles of remote PPG*, IEEE Transactions on Biomedical Engineering, vol. 64(7), s. 1479–1491. <https://doi.org/10.1109/TBME.2016.2609282>.