

Rozpoznawanie i klasyfikacja chorób skórnych z wykorzystaniem głębokich sieci neuronowych

Kacper Kołodyński i Szymon Witusiak

Politechnika Wrocławska

Słowa kluczowe: rozpoznawanie chorób skóry · głębokie sieci neuronowe

1 Wstęp

Choroby skórne stanowią powszechny problem zdrowotny na całym świecie, dotykający miliony ludzi każdego roku. Rozpoznanie i klasyfikacja chorób skórnych jest ważnym aspektem diagnostyki dermatologicznej, a skuteczne rozpoznanie wymaga nie tylko doświadczenia i wiedzy specjalisty, ale również dostępu do zaawansowanych narzędzi diagnostycznych.

W ostatnich latach, dzięki postępom w dziedzinie uczenia maszynowego i głębokich sieci neuronowych, możliwe stało się zastosowanie technologii sztucznej inteligencji do klasyfikacji chorób skórnych. W niniejszym artykule przedstawimy przegląd literatury dotyczącej zastosowania głębokich sieci neuronowych w diagnostyce chorób skórnych oraz omówimy wyniki badań nad skutecznością tych technologii w porównaniu z tradycyjnymi metodami diagnostycznymi.

W ramach badań przeprowadzonych na dużych zbiorach danych chorób skórnych wykazano, że zastosowanie głębokich sieci neuronowych pozwala na uzyskanie wysokiej skuteczności w klasyfikacji chorób skórnych, co może mieć istotne znaczenie dla poprawy diagnostyki i skuteczności leczenia tych chorób. W artykule omówimy również kwestie związane z interpretowalnością wyników uzyskanych za pomocą głębokich sieci neuronowych oraz potencjalne wyzwania związane z ich wdrożeniem w praktyce klinicznej.

W podsumowaniu przedstawimy wnioski wynikające z przeglądu literatury oraz perspektywy związane z dalszym rozwojem zastosowań sztucznej inteligencji w diagnostyce chorób skórnych.

2 Przegląd literatury

Głębokie sieci neuronowe, które są jednym z narzędzi sztucznej inteligencji, stały się obiecującą technologią w tej dziedzinie rozpoznawania obrazów. Wielu naukowców podejmuje się zastosowania tej techniki do rozpoznawania chorób. Jedną z ważniejszych prac w tej dziedzinie jest "Dermatologist-level classification of skin cancer with deep neural networks" [1]. Opisuje badania przeprowadzone przez grupę naukowców z Uniwersytetu Stanforda, którzy wykorzystali głębokie sieci neuronowe do klasyfikacji trzech najczęstszych typów raka skóry: czerniaka, raka podstawnokomórkowego i raka płaskonabłonkowego. Autorzy artykułu

porównali skuteczność klasyfikacji uzyskaną za pomocą modeli sztucznej inteligencji z wynikami uzyskanymi przez dermatologów.

W badaniach wykorzystano zbiór danych składający się z ponad 129 tysięcy zdjęć skóry, w tym 2000 zdjęć raka skóry. Następnie naukowcy trenowali modele głębokich sieci neuronowych, wykorzystując architekturę sieci Inception-v4. Wyniki klasyfikacji uzyskane przez modele porównano z wynikami uzyskanymi przez 21 doświadczonych dermatologów, którzy mieli za zadanie sklasyfikować zdjęcia w dwóch kategoriach: podejrzone i nienależące do podejrzanych.

Wyniki badań wykazały, że modele głębokich sieci neuronowych osiągnęły wyniki porównywalne z dermatologami w diagnozowaniu raka skóry. W przypadku klasyfikacji czerniaka, najlepszy model osiągnął skuteczność na poziomie 91%, co jest porównywalne z wynikami uzyskanymi przez dermatologów specjalizujących się w tej dziedzinie. W przypadku raka podstawonokomórkowego i raka płaskonabłonkowego modele osiągnęły skuteczność na poziomie 96% i 93%, odpowiednio.

W artykule autorzy podkreślają, że wyniki badań sugerują, że głębokie sieci neuronowe mogą stanowić skuteczne narzędzie w diagnozowaniu raka skóry, co może mieć istotne znaczenie dla poprawy diagnostyki i skuteczności leczenia tej choroby. Jednocześnie, autorzy artykułu zwracają uwagę na potencjalne wyzwania związane z wdrożeniem technologii sztucznej inteligencji w praktyce klinicznej, takie jak interpretowalność wyników i potrzeba weryfikacji wyników uzyskanych za pomocą modeli sztucznej inteligencji przez ekspertów.

2.1 Convolutional Neural Networks (CNN)

Konwolucyjne sieci neuronowe zostały wprowadzone w 1998 roku, przez autorów pracy [2]. Zaproponowali oni sieć neuronową z warstwami konwolucyjnymi, która potrafiła skutecznie wyodrębnić cechy z obrazów i dokonywać klasyfikacji na podstawie tych cech. Sieć ta była w stanie rozpoznawać różne cyfry i litery alfabetu łacińskiego.

Ważną cechą zaproponowanej sieci było to, że warstwy konwolucyjne oraz poolingowe pozwoliły na automatyczne wykrycie istotnych cech w obrazie, co pozwoliło na znaczne uproszczenie procesu ekstrakcji cech, który był do tej pory ręcznie projektowany przez ludzi.

Ponadto, autorzy zastosowali algorytm wstecznej propagacji błędów do uczenia sieci, który pozwalał na dostosowanie wag neuronów w celu minimalizacji błędów klasyfikacji.

Artykuł ten jest uważany za jedno z pierwszych, które zaproponowało wykorzystanie CNN w zastosowaniach przetwarzania obrazów i miał ogromny wpływ na rozwój tej dziedziny w sztucznej inteligencji.

Architektura CNN składa się z kilku warstw, które wykonują różne operacje na danych wejściowych (zazwyczaj obrazach). Typowe warstwy występujące w sieciach CNN to:

- **Warstwa konwolucyjna:** Ta warstwa wykonuje operację konwolucji, która polega na przesuwaniu filtrów (maski) po obrazie wejściowym i mnożeniu elementów maski z pikselami w odpowiednich miejscach. Operacja ta pozwala na wykrycie różnych cech w obrazie, takich jak krawędzie, kształty, tekstury, itp.
- **Warstwa aktywacji:** Po każdej warstwie konwolucyjnej następuje warstwa aktywacji, która stosuje nieliniową funkcję aktywacji do wyników z poprzedniej warstwy. Najczęściej stosowaną funkcją aktywacji jest ReLU (Rectified Linear Unit).
- **Warstwa poolingowa:** Ta warstwa wykonuje operację down-samplingu (zmniejszania rozmiaru) danych. Najczęściej stosowaną operacją jest max-pooling, która wybiera maksymalną wartość z określonego obszaru.
- **Warstwa w pełni połączona:** Ta warstwa stanowi klasyczną sieć neuronową z pełnymi połączeniami. Każdy neuron z poprzedniej warstwy jest połączony ze wszystkimi neuronami w tej warstwie. Ta warstwa zwykle stanowi końcową część sieci i dokonuje klasyfikacji.

Współczesne architektury CNN, takie jak VGG, ResNet czy Inception, składają się z kilku powtarzających się bloków, które składają się z kilku warstw konwolucyjnych, aktywacji i poolingowych. Takie modele mają bardzo wiele parametrów, co pozwala na uzyskanie bardzo wysokiej dokładności w zadaniach przetwarzania obrazów, ale także wymagają dużych zasobów obliczeniowych i danych treningowych.

3 Plan eksperymentów

- Przetwarzanie i przygotowanie danych: Wstępne przetworzenie danych, np. skalowanie, normalizacja. Podział danych na zbiór treningowy, walidacyjny i testowy.
- Trenowanie sieci z wykorzystaniem danych ze zbioru treningowego. Zostanie przeprowadzone kilka eksperymentów zmieniając parametry uczenia takie jak współczynnik uczenia, liczba epok, itd. a
- Ocena jakości klasyfikacji dla każdej z sieci neuronowych wyszkolonej z innymi parametrami, wykorzystując dane ze zbioru walidacyjnego i testowego. Do oceny jakości klasyfikacji można wykorzystane zostanie miara ballanced accuracy score.
- Analiza statystyczna wyników eksperymentów z wykorzystaniem testów statystycznych, porównanie wyników dla każdej z sieci i wybór tej, która daje najlepsze wyniki.
Przy planowaniu eksperymentów ważne jest również uwzględnienie innych czynników, takich jak hiperparametry sieci, sposób walidacji krzyżowej, czy sposób regularyzacji modelu.

4 Protokół eksperymentalny

Celem eksperymentu jest porównanie wydajności sieci neuronowych wytrenowanych z różnymi parametrami w rozpoznawaniu i klasyfikacji chorób skórnych.

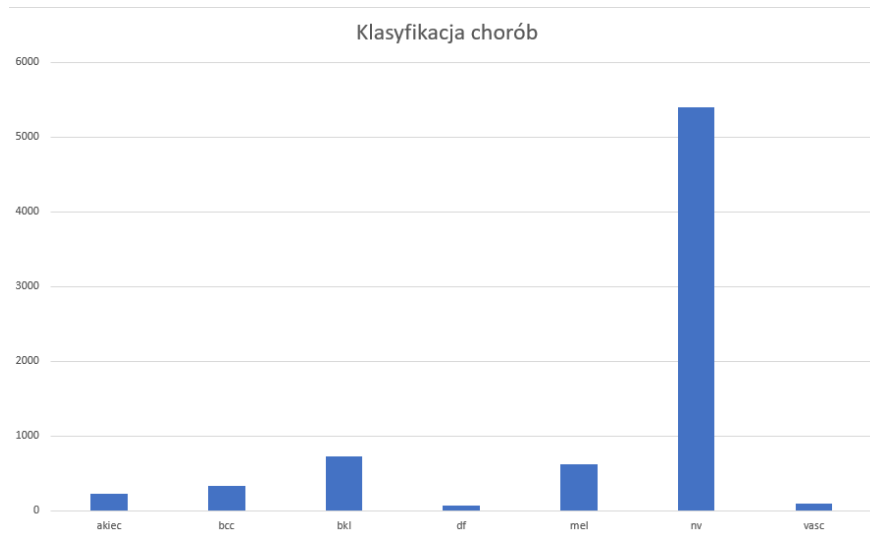
- **Dane:** Do eksperymentu zostanie wykorzystany zbiór danych Skin Cancer MNIST: HAM10000, który zawiera 10 000 zdjęć chorób skórnych zebranych z różnych źródeł.
- **Przygotowanie danych:** Dane zostaną przetworzone i przygotowane do uczenia sieci neuronowych. Wszystkie zdjęcia zostaną skalowane i znormalizowane do rozmiaru 224x224 pikseli. Zdjęcia zostaną podzielone na zbiór treningowy (70%), zbiór walidacyjny (15%) i zbiór testowy (15%).
- **Parametry uczenia sieci neuronowych:** Zostaną ustalone wartości parametrów, dla których zostaną wytrenowane sieci. Parametry zmieniane podczas eksperymentu to współczynnik uczenia (learning rate), liczba epok (epochs), batch size - współczynnik określający, ile przykładów treningowych należy użyć do aktualizacji wag w jednej iteracji. Zmieniane będą też funkcje aktywacji oraz optymalizator - algorytm, który określa, jakie zmiany powinny być wprowadzane do wag sieci neuronowej podczas procesu uczenia. Każda z sieci zostanie wytrenowana na zbiorze treningowym i oceniona na zbiorze walidacyjnym.
- **Ocena wyników:** Wyniki klasyfikacji będą oceniane na podstawie dokładności, precyzji, czułości, F1-score i krzywej ROC dla każdej z architektur. Wyniki będą porównywane i analizowane w celu określenia najlepszej architektury.
- **Wyniki:** Wnioski zostaną wyciągnięte na podstawie wyników eksperymentów i porównania wydajności różnych architektur sieci neuronowych. Porównanie wyników pomoże zidentyfikować najlepszą architekturę sieci neuronowej do klasyfikacji chorób skórnych.

4.1 Zbiór danych

Skin Cancer MNIST: HAM10000[3] to zbiór danych medycznych zawierający ponad 10 000 zdjęć zmian skórnych, w tym nowotworów skóry. Zdjęcia zostały zebrane w trakcie badań dermatologicznych i podzielone na 7 klas chorób skóry, w tym: czerniak, łuszczyca, znamiona, róży kielichowatej, podpalenie słoneczne, rogowacenie słoneczne i łagodne zmiany skórne.

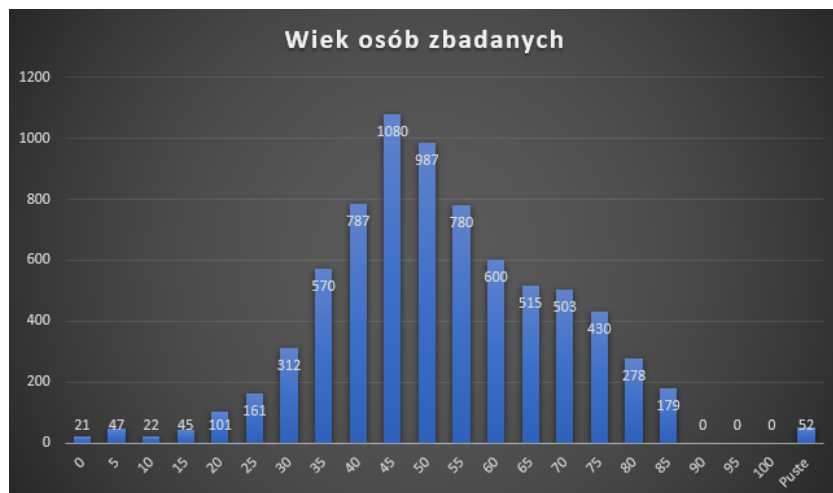
Zbiór danych Skin Cancer MNIST: HAM10000 został opracowany z myślą o wykorzystaniu w zadaniach klasyfikacji chorób skóry z wykorzystaniem techniki sztucznej inteligencji i uczenia maszynowego, w tym głębokich sieci neuronowych. Zbiór ten ma na celu ułatwienie diagnozowania chorób skórnych oraz poprawienie efektywności i skuteczności terapii.

Zdjęcia wchodzące w skład zbioru danych Skin Cancer MNIST: HAM10000 zostały poddane procesowi standaryzacji, aby umożliwić porównywanie wyników badań i analizę skuteczności różnych algorytmów klasyfikacji. Ponadto, zbiór ten jest dostępny publicznie, co umożliwia naukowcom i badaczom z różnych dziedzin korzystanie z tych danych i rozwijanie nowych algorytmów klasyfikacji chorób skóry.



Rysunek 1. Ilość wystąpień poszczególnych etykiet w całym zbiorze danych

Powyższy wykres ilustruje duże niebalansowanie w zbiorze danych. W badaniach dodatkowych polegających na przetestowaniu modeli na bardziej zbalansowanym zbiorze danych ograniczono instancje klasy "nv", a przypadki chorób występujących pod klasami "vasc" oraz "akiec" usunięto.



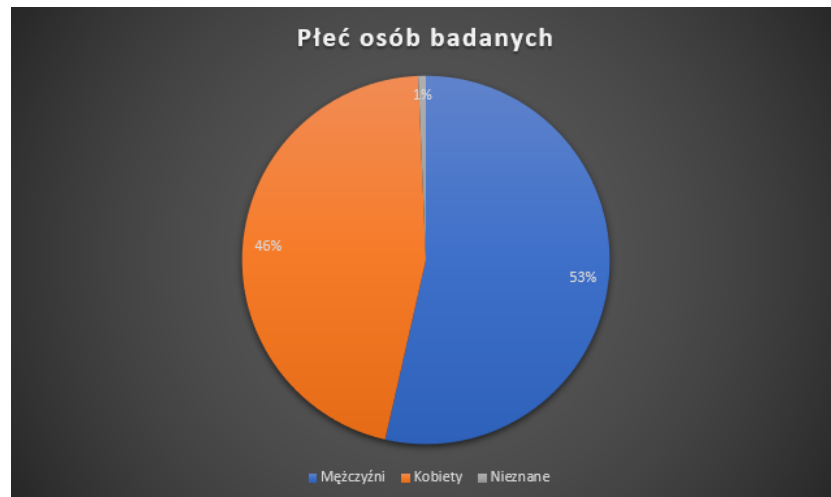
Rysunek 2. Wiek osób przebadanych

Osoby, których znamiona zostały przedstawione na wykorzystywanych zdjęciach w głównej mierze są osobami z przedziału: 35-70 (lat).



Rysunek 3. Miejsca występowania znamion

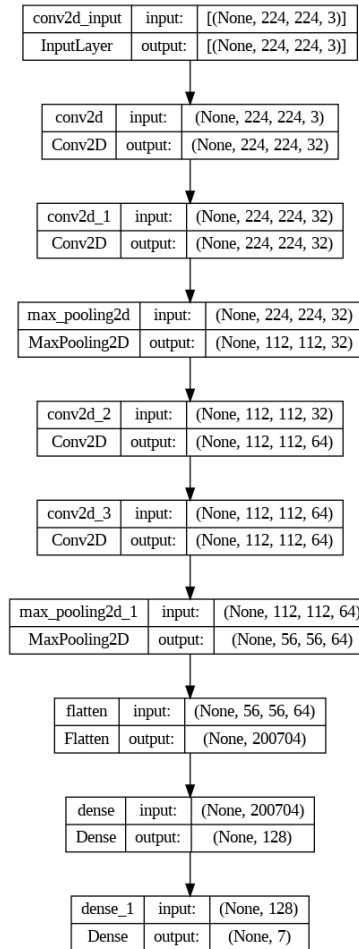
Rysunek 3 obrazuje dominację pewnych obszarów na ciele, jeśli chodzi o występowanie chorób skóry.



Rysunek 4. Wykres płci osób badanych

Powyższe informacje są dodatkowymi danymi odnośnie zestawu danych, jednakże nie zostały one wykorzystane oraz nie przyczyniły się w żaden sposób do trenowania modelu.

5 Wyniki eksperymentów



Rysunek 5. Architektura własnej sieci

Epochs \ Batch size	Batch size			
	8	16	32	64
30	0.287	0.411	0.391	0.443
50	0.143	0.407	0.360	0.247

Tabela 1. Wyniki *balanced accuracy* dla zbioru testowego

Batch size	8	16	32	64
Epochs				
30	1.235	1.252	1.632	0.845
50	1.081	1.817	1.032	1.030

Tabela 2. Wyniki *loss* dla zbioru testowego

5.1 Test sumy rang Wilcoxona

Do sprawdzenia czy różnice między wynikami są istotne statystycznie wykorzystano test sumy rang Wilcoxona - *ranksums* z biblioteki *scipy.stats*.

Test *ranksums* to nieparametryczny test statystyczny używany do porównywania dwóch niezależnych próbek. Test ten opiera się na porównaniu sum rang dwóch próbek, które reprezentują wartości obserwowane w dwóch grupach.

Podczas przeprowadzania testu *ranksums* wylicza się *statystykę testową* i wartość p , które wskazują na istnienie różnic między dwiema grupami. Jeśli wartość p jest mniejsza niż ustalony poziom istotności, to można odrzucić hipotezę zerową i stwierdzić, że istnieje istotna statystycznie różnica między grupami.

5.2 Porównanie wyników w zależności od liczby epok

Dla porównania liczby epok **30** i **50** uzyskano wartość statystyki na poziomie **2.309** i wartość p **0.021**.

Wynika z tego że dla przyjętego *poziomu istotności* $p = 0.05$ - wyniki są statystycznie różne - wyniki dla **30** epok są statystycznie lepsze.

Wyniki funkcji straty okazały się nieistotne statystycznie.

5.3 Porównanie wyników *balanced accuracy* w zależności od *batch size*

Batch size	8	16	32	64
8	0.000	-1.549	-0.549	-0.775
16	1.549	0.000	1.549	0.000
32	1.549	-1.549	0.000	0.000
64	0.775	0.000	0.000	0.000

Tabela 3. Wyniki *statystyki* dla zbioru testowego

Batch size	8	16	32	64
8	1.000	0.121	0.121	0.439
16	0.121	1.000	0.121	1.000
32	0.121	0.121	1.000	1.000
64	0.439	1.000	1.000	1.000

Tabela 4. Wyniki *p-value* dla zbioru testowego

Dla przyjętego *poziomu istotności* $p = \mathbf{0.05}$ - wyniki nie różnią się statystycznie w zależności od wielkości *batch size*.

5.4 Porównanie wyników *balanced accuracy* w zależności od *liczby epok*

Epochs	30	50
30	0.000	1.443
50	-1.443	0.000

Tabela 5. Wyniki *statystyki* dla zbioru testowego

Epochs	30	50
30	0.000	0.149
50	0.149	0.000

Tabela 6. Wyniki *p-value* dla zbioru testowego

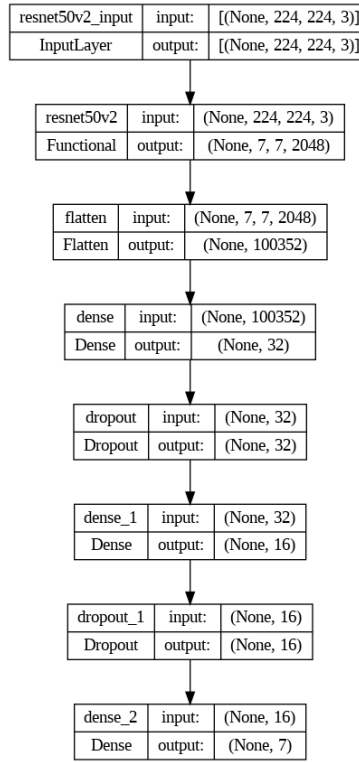
Dla przyjętego *poziomu istotności* $p = \mathbf{0.05}$ - wyniki nie różnią się statystycznie w zależności od wielkości *liczby epok*.

Poniższe eksperymenty wykonane na sieci VGG oraz Resnet50V2 zostały wykonane przy parametrach *batch size* = 16 i *epochs* = 50

5.5 ResNet

Przeprowadzono również eksperymenty na sieci o architekturze *ResNet* [7].

Dla modelu sieci o architekturze:



Rysunek 6. Architektura zawierająca sieć ResNet, konfiguracja 1

uzyskano wyniki:

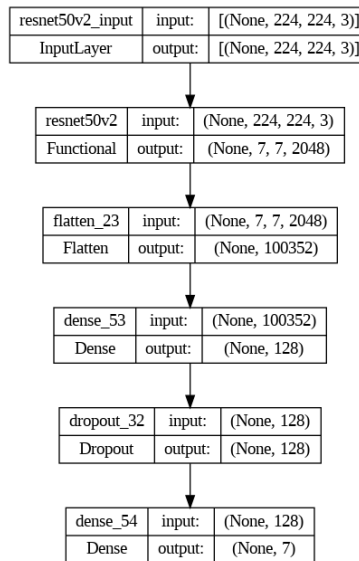
Batch size		16	32
Epochs	50	0.143	0.143

Tabela 7. Wyniki *balanced accuracy* dla zbioru testowego

Batch size		16	32
Epochs	50	1.375	1.211

Tabela 8. Wyniki *loss* dla zbioru testowego

Dla modelu sieci o architekturze:



Rysunek 7. Architektura zawierająca sieć ResNet, konfiguracja 2

uzyskano wyniki:

Epochs	Batch size	
	16	32
50	0.153	0.152

Tabela 9. Wyniki *balanced accuracy* dla zbioru testowego

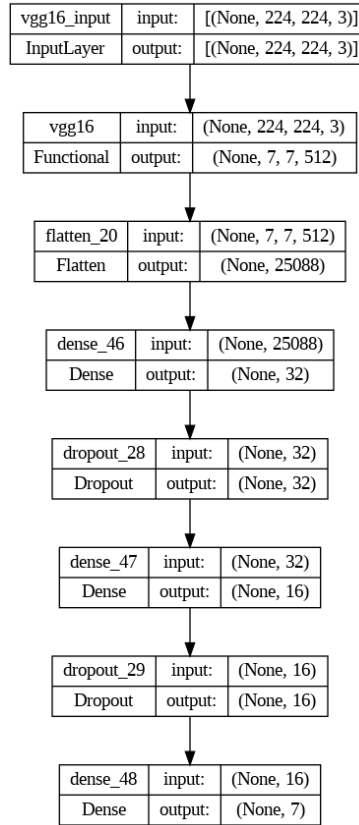
Epochs	Batch size	
	16	32
50	1.387	1.358

Tabela 10. Wyniki *loss* dla zbioru testowego

5.6 VGG (Visual Geometry Group)

Przeprowadzono również eksperymenty na sieci o architekturze *VGG*[6].

Dla modelu sieci o architekturze:



Rysunek 8. Architektura zawierająca sieć VGG16, konfiguracja 1

uzyskano wyniki:

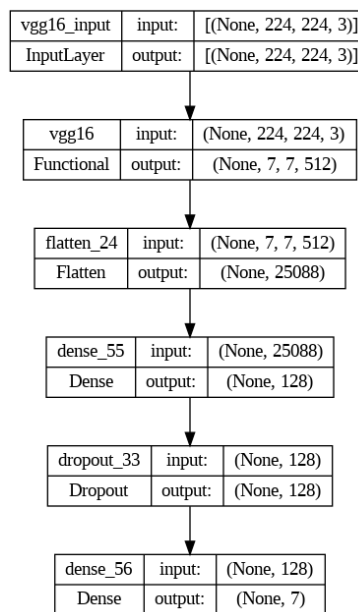
Batch size		16	32
Epochs			
50		0.428	0.178

Tabela 11. Wyniki *balanced accuracy* dla zbioru testowego

Batch size		16	32
Epochs			
50		1.618	1.136

Tabela 12. Wyniki *loss* dla zbioru testowego

Dla modelu sieci o architekturze:



Rysunek 9. Architektura zawierająca sieć VGG16, konfiguracja 2

uzyskano wyniki:

Batch size		16	32
Epochs	50	0.328	0.304

Tabela 13. Wyniki *balanced accuracy* dla zbioru testowego

Batch size		16	32
Epochs	50	1.244	1.272

Tabela 14. Wyniki *loss* dla zbioru testowego

6 Wyniki eksperymentów na zmodyfikowanym zbiorze danych

Z racji, że wyniki wykonanych eksperymentów nie spełniły oczekiwań, postanowiono zmodyfikować zestaw danych. Wykorzystany zestaw danych był bardzo niebalansowany, dominująca klasa posiadała ponad 7 tysięcy rekordów, gdzie dwie klasy mniejszościowe nieco poniżej 100. Ze względu na ogromne różnice między występowaniem instancji danych klas, wytrenowane modele miały problem z klasyfikacją. Dlatego też wykonano następujące eksperymenty z wykorzystaniem różnych modeli sieci neuronowych dla konfiguracji numer 1:

1. Ograniczenie klasy dominującej do 1000 instancji oraz usunięcie dwóch klas mniejszościowych
2. Ograniczenie klasy dominującej do 500 instancji oraz usunięcie dwóch klas mniejszościowych

6.1 Eksperyment 1

Resnet50V2 - Zbiór walidacyjny:
 Dokładność: 0.677596
 "Loss": 0.811289

Zbiór testowy:
 Zrównoważona dokładność: 0.499494
 "Loss": 1.319659

Vgg16 - Zbiór walidacyjny:
 Dokładność: 0.568306
 "Loss": 1.132107

Zbiór testowy:
 Zrównoważona dokładność: 0.371429
 "Loss": 1.081733

Własna sieć - Zbiór walidacyjny:
 Dokładność: 0.693989
 "Loss": 0.747669

Zbiór testowy:
 Zrównoważona dokładność: 0.442335
 "Loss": 2.130459

6.2 Eksperyment 2

Resnet50V2 - Zbiór walidacyjny:

Zrównoważona dokładność: 0.442335
"Loss": 0.983731

Zbiór testowy:

Zrównoważona dokładność: 0.543117
"Loss": 1.820771

Vgg16 - Zbiór walidacyjny:

Dokładność: 0.403846
"Loss": 1.463677

Zbiór testowy:

Zrównoważona dokładność: 0.251847
"Loss": 1.326200

Własna sieć - Zbiór walidacyjny:

Dokładność: 0.500000
"Loss": 1.510978

Zbiór testowy:

Zrównoważona dokładność: 0.334478
"Loss": 4.960614

6.3 Porównanie wyników *balanced accuracy* w zależności od zbioru uczącego

Zbiór	początkowy	zbalansowany 1	zbalansowany 2
początkowy	0.000	-1.964	-0.655
zbalansowany 1	1.964	0.000	0.655
zbalansowany 2	0.655	-0.655	0.000

Tabela 15. Wyniki *statystyki* dla zbioru testowego

Zbiór	początkowy	zbalansowany 1	zbalansowany 2
początkowy	1.000	0.050	0.513
zbalansowany 1	0.050	1.000	0.513
zbalansowany 2	0.513	0.513	1.000

Tabela 16. Wyniki *p-value* dla zbioru testowego

Zbiór *zbalansowany 1* jest istotnie statystycznie lepszy od zbioru początkowego, pod względem uzyskanej *balanced accuracy* dla trzech powyższych sieci neuronowych ($p\text{-value} \leq 0.05$).

7 Wnioski

Uzyskane wyniki dokładności klasyfikacji dla zbioru testowego sięgają 74%. Ani *liczba epok*, ani *batch size* nie są kluczowymi parametrami wpływającymi na dokładność klasyfikacji.

W badaniach nie osiągnięto pożądaných efektów poprzez wystąpienie "overfittingu", model stracił zdolność do generalizacji informacji przez co jego predykcje nie były tak skuteczne jak mogłyby być. Własna sieć pomimo, że osiągnęła wyniki stosunkowo przeciętne, wymaga ponownego dostosowania hiperparametrów, w celu uzyskania lepszej jakości. Poza wartościami hiperparametrów takich jak wartość porzucenia, duże znaczenie ma wykorzystany zestaw danych. Jest on mocno niezbalansowany, do tego stopnia, że jedna klasa posiada około 70% instancji, gdzie aż trzy typy występowania choroby oscylują w okolicach 5%. Z tego względu model ma problem z generalizacją cech a co za tym idzie z klasyfikacją. Potencjalnym rozwiązaniem tego problemu mogłaby być augmentacja danych, rzadziej występujących klas aby swoją liczebnością były zbliżone do klasy dominującej. Niedokładności w klasyfikacji raka skóry mogą być związane z dominacją klasy większościowej oraz zaniżeniem wydajności klasyfikatora dla rzadszych klas, zwłaszcza czerniaka.

Wnioskuje się, że niezbalansowanie zbioru danych jest jednym z głównych czynników, które przyczyniają się do tych niedokładności. Dlatego też powtórzono jeden z eksperymentów na zestawie danych, która został w pewnym stopniu zbalansowany. Dzięki pomniejszeniu klasy dominującej modele były w stanie lepiej przyswoić dane, a przy tym lepiej generalizować, co wpływało na klasyfikację zdjęć. Bardzo dobrze widoczne było to na otrzymanych macierzach pomyłek, które pokazały, że klasyfikacja nie skupia się na jednej klasie tylko na wielu. W przypadku niezbalansowanego zbioru danych bardzo duża część predykcji polegała na pozytywnym przypisaniu etykiety to klasy dominującej, ponieważ informacje o niej najlepiej przyswoiły modele. W momencie gdy ta klasa została ograniczona, a dwie klasy, które posiadały bardzo małą ilość instancji w porównaniu do innych klas, zostały usunięte, model zaczął polepszać swą skuteczność w przypisywaniu chorób do zdjęć innych schorzeń. Precyzja modelu nieco spadła, jednakże jest to spowodowane faktem, że przedtem modele poprawnie klasyfikowały jedną chorobę, która stanowiła około 70% zbioru. Gdy ta klasa została w pewnym stopniu wyrównana z innymi występowaniami klas, model stracił skuteczność, jednakże zaczął lepiej klasyfikować pozostałe choroby. Dlatego też można stwierdzić, że ograniczenie klasy dominującej oraz odcięcie dwóch klas z małą ilością przypadków, aby model dobrze je rozpoznawał, wpłynęły pozytywnie na trening modeli. Oczywiście na tak zmodyfikowanym zestawie danych warto byłoby, powtórzyć inne eksperymenty, które zapobiegłyby przeuczeniu, w ten sposób jakość modele by uległa polepszeniu, tym samym udałoby się stworzyć model z praktycznym wykorzystaniem.

Literatura

1. Andre esteva, Brett Kuprel, Roberto A. Novoa, Justin Ko, Susan M. Swetter, Helen M. blau, Sebastian Thrun, Dermatologist-level classification of skin cancer with deep neural networks. 2017.
2. Yann LeCun, Léon Bottou, Yoshua Bengio, Patrick Haffner, Gradient-Based Learning Applied to Document Recognition. 1998.

3. Tschandl, P., Rosendahl, C., Kittler, H., The HAM10000 dataset: a large collection of multi-source dermatoscopic images of common pigmented skin lesions. Scientific data. 2018.
4. Rosendahl, Cliff and Tschandl, Philipp and Cameron, Alan and Kittler, Harald, Diagnostic accuracy of dermatoscopy for melanocytic and nonmelanocytic pigmented lesions, pages 1068-1073, 2011.
5. Tschandl, Philipp and Rinner, Christoph and Apalla, Zoe and Argenziano, Giuseppe and Codella, Noel and Halpern, Allan and Janda, Monika and Lallas, Aimilios and Longo, Caterina and Malvehy, Josep and others, Human-computer collaboration for skin cancer recognition, pages 1229–1234, 2020.
6. Theckedath, R. Sedamkar. Detecting affect states using vgg16, resnet50 and seresnet50 networks. SN Computer Science, 1:1–7, 2020.
7. Rahimzadeh, A. Attar. A modified deep convolutional neural network for detecting covid-19 and pneumonia from chest x-ray images based on the concatenation of xception and resnet50v2. Informatics in medicine unlocked, 19:100360, 2020.