**Politechnika Wrocławska**

**Wydział Informatyki i Telekomunikacji**

# Kierunek: inżynieria systemów (INS)

## PRACA DYPLOMOWA

## MAGISTERSKA

**Opracowanie algorytmu klasyfikacji rentgenogramów**

**w infekcji COVID-19**

Imię i nazwisko dyplomanta

Jakub Zolich

Opiekun pracy

**Dr hab. inż. Krzysztof Brzostowski**

Słowa kluczowe: algorytm, klasyfikacja, COVID-19, sieci neuronowe

### WROCŁAW (2024)

# Streszczenie

# Abstract

Spis treści

[1. Wprowadzenie 5](#_Toc156313349)

[1.1. Geneza pracy 5](#_Toc156313350)

[1.2. Przegląd znanych rozwiązań 7](#_Toc156313351)

[1.2.1. RT-PCR (Real-Time Polymerase Chain Reaction) 7](#_Toc156313352)

[1.2.2. Testy antygenowe 7](#_Toc156313353)

[1.2.3. Testy serologiczne 8](#_Toc156313354)

[1.2.4. Podsumowanie obecnych rozwiązań 9](#_Toc156313355)

[1.3. Narzędzia 10](#_Toc156313356)

[1.3.1. Stos technologiczny 10](#_Toc156313357)

[1.3.2. Podstawowe modele i metody uczenia maszynowego 11](#_Toc156313358)

[1.3.3. Zbiór danych 15](#_Toc156313359)

[1.4. Cel pracy 15](#_Toc156313360)

[1.5. Zakres prac 16](#_Toc156313361)

[2. Projekt algorytmu klasyfikacji 18](#_Toc156313362)

[2.1. Analiza wymagań funkcjonalnych i niefunkcjonalnych 18](#_Toc156313363)

[2.2. Architektura algorytmu 19](#_Toc156313364)

[3. Implementacja algorytmu 19](#_Toc156313365)

[4. Prezentacja algorytmu 19](#_Toc156313366)

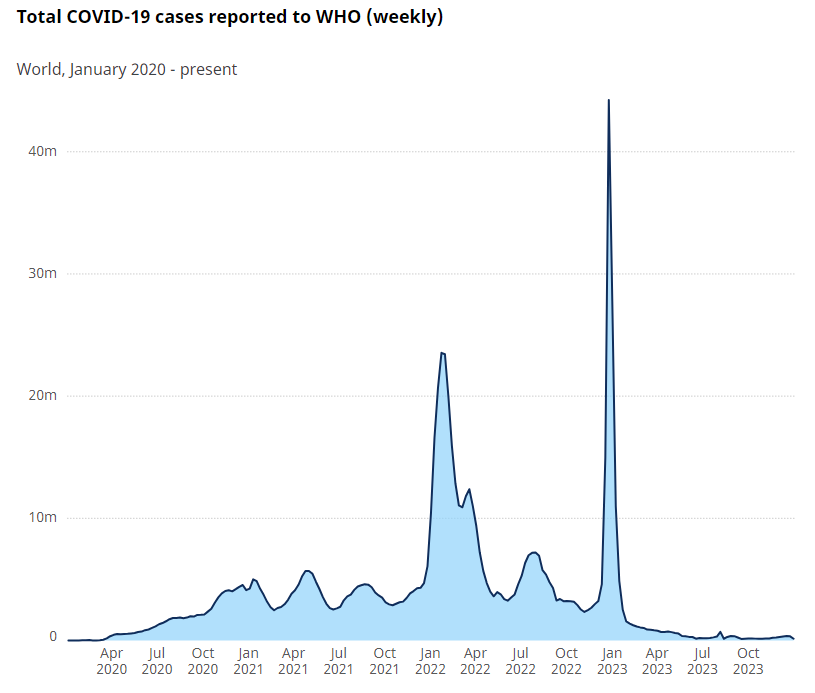
[5. Podsumowanie 19](#_Toc156313367)

[6. Bibliografia 19](#_Toc156313368)

# Wprowadzenie

## Geneza pracy

Kryzys zdrowia publicznego wywołany przez pandemię COVID-19 przyniósł wiele wyzwań dla systemów opieki zdrowotnej na całym świecie. Rentgenogramy płuc są często wykorzystywane jako narzędzie diagnostyczne do wykrywania i monitorowania przebiegu choroby. Jednakże, ich interpretacja może być skomplikowana i czasochłonna, wymaga bowiem zaangażowania wysoko wykwalifikowanego personelu medycznego. W obliczu tych wyzwań, potrzeba nowych narzędzi i algorytmów, które umożliwią automatyczną klasyfikację rentgenogramów płuc po infekcji wirusem COVID-19, stała się pilna.



Radiografia klatki piersiowej (CXR) jest jednym z najczęściej używanych narzędzi diagnostycznych w radiologii ze względu na jej znaczenie dla zdrowia publicznego, powszechne zastosowanie i stosunkowo niski koszt. Obrazowanie klatki piersiowej, zarówno tomografia komputerowa (CT) jak i CXR, odegrało kluczową rolę w diagnozowaniu i monitorowaniu postępu choroby COVID-19. Wiele badań skupiło się na opracowaniu algorytmów opartych na głębokim uczeniu do przetwarzania obrazów CXR, takich jak diagnostyka chorób klatki piersiowej.

Nie mniej jednak, istniejące metody oparte na uczeniu maszynowym często zmagają się z problemami związanymi z jakością danych treningowych, a w przypadku danych medycznych, etykiety są często zaszumione lub niekompletne, co negatywnie wpływa na wydajność modeli klasyfikacji. Ponadto, interpretacja wyników obrazowania płuc jest wyzwaniem nawet dla doświadczonych radiologów, zwłaszcza w kontekście nowego patogenu.

Różne badania wykazały doskonałą wydajność metod głębokiego uczenia w ekstrakcji cech obrazu od niskiego do wysokiego poziomu i uczenia reprezentacji dyskryminacyjnych (tj. embeddingów) z dużych ilości danych. Jako jedna z najczęstszych modalności obrazowania dla diagnostycznych badań radiologicznych, radiogram klatki piersiowej (CXR) cieszy się ogromną uwagą w dziedzinie analizy obrazu opartej na sztucznej inteligencji ze względu na jej znaczenie dla zdrowia publicznego, szerokie wykorzystanie i stosunkowo niski koszt. Przeprowadzono szereg badań dotyczących przetwarzania obrazów dla CXR z wykorzystaniem głębokiego uczenia, m.in. diagnostykę chorób klatki piersiowej.

Obrazowanie klatki piersiowej, w tym tomografia komputerowa (CT) i CXR, odgrywało kluczową rolę w diagnozowaniu i monitorowaniu rozprzestrzeniania się wirusa. Na przykład, gdy podaż i dokładność testów COVID-19 w reakcji łańcuchowej polimerazy (PCR), nie mogła zaspokoić potrzeb klinicznych, CXR klatki piersiowej została zalecona jako narzędzie przesiewowe w wytycznych dotyczących zarządzania COVID-19 podczas wczesnego wybuchu choroby w Wuhan. W szczególności, społeczność zajmująca się analizą obrazów medycznych szybko zareagowała, opracowując nowatorskie rozwiązania diagnostyczne i segmentacyjne COVID-19, w tym prace, w których bardzo wysoką specyficzność uzyskano za pomocą 3D Resnet. Radiografia klatki piersiowej, zwłaszcza przenośne urządzenia radiograficzne, są uważane za medycznie niezbędne w placówkach opieki ambulatoryjnej, ponieważ nie wymagają przeniesienia pacjenta do działu obrazowania i są łatwiejsze do sterylizacji. Wraz z większą ilością zdjęć CXR COVID-19, można zaobserwować i podsumować spójne ustalenia, takie jak zacienienia matowej szyby w obu płucach. Ustalenia te sugerują potencjał wykorzystania CXR do oceny ciężkości choroby (na podstawie całkowitego zajęcia płuc), monitorowania progresji choroby i przewidywania rokowania pacjenta. Jednak nadal wyzwaniem nawet dla doświadczonych radiologów jest interpretacja tych niespecyficznych wyników z pewnością, zwłaszcza na CXR, ponieważ istnieje wiele niewiadomych na temat nowej choroby zakaźnej.

Szybka i dokładna klasyfikacja obrazów CXR może znacznie przyspieszyć proces diagnostyczny, zwłaszcza w miejscach o ograniczonym dostępie do wysoko wykwalifikowanego personelu medycznego. Nowe algorytmy mają potencjał do łagodzenia obciążenia systemów opieki zdrowotnej poprzez zautomatyzowanie niektórych aspektów diagnostyki radiologicznej, co umożliwia lekarzom skupienie się na najtrudniejszych przypadkach. Dalsze badania nad zastosowaniami algorytmów w innych dziedzinach medycyny, jak również jego integracja z istniejącymi systemami medycznymi, staną się kluczowym krokiem w kierunku pełnego wykorzystania potencjału, jakie niesie ze sobą sztuczna inteligencja w opiece zdrowotnej.

Dlatego system AI, który może uczyć się od najlepszych radiologów i zapewniać spójne wyniki, byłby bardzo cenny w praktyce klinicznej. W odpowiedzi na niedobór radiologów w obsłudze obrazów CXR, zwłaszcza w krajach rozwijających się, można wykorzystać narzędzia diagnostyczne COVID-19 wspomagane przez AI.

## Przegląd znanych rozwiązań

Dział 1.2. omawia różne metody diagnostyczne stosowane do wykrywania COVID-19. Skupiono się na trzech głównych kategoriach testów: testach kwasów nukleinowych (w tym RT-PCR), testach antygenowych oraz testach serologicznych. Każda z tych metod charakteryzuje się unikalnymi mechanizmami działania, zaletami oraz ograniczeniami. Testy kwasów nukleinowych, znane również jako RT-PCR, opierają się na wykrywaniu materiału genetycznego wirusa, podczas gdy testy antygenowe skupiają się na identyfikacji obcych cząsteczek na powierzchni wirusa. Z kolei testy serologiczne analizują odpowiedź immunologiczną organizmu na obecność wirusa, poprzez wykrywanie przeciwciał. W tym rozdziale dokładnie przyjrzano się każdej z tych metod, ich zastosowaniu w diagnostyce, czasie oczekiwania na wyniki oraz warunkom, w jakich najlepiej je stosować, aby zapewnić jak najdokładniejsze i najbardziej użyteczne informacje na temat obecności i ewolucji wirusa SARS-CoV-2 w populacji.

### Testy diagnostyczne i immunologiczne

1. Testy RT-PCR

Testy kwasów nukleinowych, RT-PCR i testy molekularne to te same nazwy dla tego testu. Sprawdza on obecność materiału genetycznego wewnątrz wirusa wywołującego COVID-19. DNA i RNA to kwasy nukleinowe zwane również materiałem genetycznym. PCR to skrót od reakcji łańcuchowej polimerazy. Polimeraza to enzym, który może tworzyć kopie materiału genetycznego. Dlaczego potrzebujemy PCR?

Materiał genetyczny można wyizolować w laboratorium, jednak tylko niewielką jego ilość można wyizolować ze zwierzęcia, rośliny lub wirusa. PCR służy do tworzenia wielu kopii tego materiału genetycznego, aby można go było zbadać. RT w RT-PCR oznacza odwrotną transkryptazę. Odwrotna transkryptaza odczytuje nić RNA i tworzy pasującą kopię DNA. Wirus wywołujący COVID-19 jest wirusem RNA. RT-PCR zamienia RNA wirusa w DNA, a następnie tworzy wiele kopii DNA. Możemy wtedy sprawdzić, czy RNA wirusa było obecne, jeśli miała miejsce PCR, obecne jest dużo DNA. To DNA może być później sekwencjonowanie, patrząc na każdą parę zasad kwasu jądrowego i porównywane z innymi sekwencjami wirusa. W ten sposób wiemy, że wirus się zmienił i kiedy pojawiają się nowe warianty.

Zaletą tego testu jest to, że można go wykonać w dowolnym momencie, nawet jeśli dana osoba nie doświadcza objawów. Wadą jest to, że wymaga laboratorium i specjalnego sprzętu. Inną wadą jest to, że polimeraza potrzebuje czasu, aby wykonać wiele kopii DNA. Zazwyczaj próbki są pobierane w klinice, a następnie wysyłane do laboratorium.

Zalety testu RT-PCR:

* Dokładny, nawet jeśli dana osoba nie ma objawów
* Może być stosowany do diagnozowania COVID-19

Wady testu RT-PCR:

* Wymaga laboratorium, nie może być wykonany w domu
* Zwykle trwa 24-72 godziny, 1-3 dni w zależności od laboratorium

1. Testy antygenowe

Antygen to obca cząsteczka obecna na powierzchni wirusa. Kiedy nasz organizm jest narażony na działanie wirusów, takich jak koronawirus, aktywuje mechanizmy obronne w celu zwalczania antygenów wirusowych. Nasz układ odpornościowy wytwarza przeciwciała skierowane przeciwko antygenom. W tym przypadku kaseta szybkiego testu antygenowego jest wstępnie pokryta przeciwciałami, a gdy antygeny obecne w naszej próbce wchodzą w kontakt z przeciwciałami, następuje zmiana koloru na kasecie.

Testy antygenowe są idealne dla osób w pierwszych dniach objawów. Kolejną zaletą jest to, że testy antygenowe nie wymagają laboratorium do uzyskania wyników i można je wykonać w dowolnym miejscu. Testy antygenowe doskonale nadają się do badań przesiewowych. Testy antygenowe to szybkie testy, które dają wyniki w ciągu 10-30 minut w zależności od urządzenia testowego. Większość testów antygenowych wymaga 15 minut na uzyskanie wyników. Jedną z wad testów antygenowych jest ich niższa dokładność w przypadku osób, które nie mają objawów. Jeśli ktoś uzyska pozytywny wynik, zazwyczaj wykonuje się później test RT-PCR w celu potwierdzenia wyniku.

Zalety szybkiego testu antygenowego:

* Wyniki w ciągu 15 minut
* Może być wykonany w dowolnym miejscu, w tym w domu
* Może być stosowany do diagnozowania COVID-19

Wady szybkiego testu antygenowego:

* Bardziej dokładny w ciągu pierwszych 5 dni od wystąpienia objawów niż testowanie bez objawów

1. Testy serologiczne

Ludzie mają adaptacyjny układ odpornościowy. Nasze ciała wytwarzają przeciwciała przeciwko rzeczom, które nasz układ odpornościowy postrzega jako zagrożenie. Jeśli ktoś wytwarza przeciwciała przeciwko wirusowi, oznacza to, że był na niego narażony. Testy na obecność przeciwciał sprawdzają, czy organizm wytwarza przeciwciała. Wytworzenie przeciwciał po zakażeniu zajmuje 1-2 tygodnie. Jeśli więc właśnie zostałeś zarażony, test na obecność przeciwciał może dać wynik fałszywie ujemny. Test na obecność przeciwciał może również dać wynik pozytywny, jeśli badana osoba niedawno otrzymała szczepionkę. Organizm ludzki wytwarza przeciwciała w odpowiedzi na szczepionkę (tak działają szczepionki). Testy na obecność przeciwciał są zwykle wykonywane wyłącznie przez naukowców i badaczy zdrowia publicznego. Jest to świetny sposób na sprawdzenie, gdzie wirus występował w populacji w przeszłości. Nie jest to jednak dobre narzędzie diagnostyczne.

Zalety laboratoryjnego testu na obecność przeciwciał:

* Możliwość wykrycia osób, które wcześniej chorowały na COVID-19

Wady laboratoryjnego testu na obecność przeciwciał:

* Wymaga laboratorium. Nie może być wykonany w domu
* Zwykle trwa 24-72 godziny, 1-3 dni w zależności od laboratorium
* Nie może być używany do diagnozowania COVID-19
* Wykonywany 3-4 tygodnie po zakażeniu
* Szczepienie może spowodować fałszywie dodatni wynik, w zależności od techniki testowania

### Podsumowanie testów diagnostycznych i immunologicznych

|  |  |  |  |
| --- | --- | --- | --- |
|  | RT – PCR | Antygenowy | Serologiczny |
| Czas oczekiwania na wyniki | 24-72 godziny | 10 – 30 min | 24-72 godziny |
| Kiedy powinno się wykonać test | Kiedykolwiek | Pierwsze 5 dni od wystąpienia symptomów | 3 – 4 tygodnie po wystąpieniu objawów |
| Zastosowanie diagnostyczne | Tak | Tak | Nie |
| Lokalizacja | W domu, samodzielnie | Laboratorium | Laboratorium |
| Typ próbki | Wymaz, krew, ślina | Wymaz, krew, ślina | Krew |
| Używane przez | Ogólną populację, specjalistów medycznych | Ogólną populację, specjalistów medycznych | Badaczy |

Testy RT-PCR, uznawane za złoty standard w diagnostyce, wykazują wysoką dokładność w wykrywaniu materiału genetycznego wirusa, lecz są czasochłonne i wymagają specjalistycznych laboratoriów. Testy antygenowe zapewniają szybsze wyniki i są bardziej dostępne, ale ich dokładność jest niższa, zwłaszcza u osób bez objawów. Testy serologiczne, użyteczne w badaniach epidemiologicznych, identyfikują obecność przeciwciał, wskazując na wcześniejszą ekspozycję na wirusa, jednak nie są skuteczne w wykrywaniu aktywnej infekcji. Podsumowując, każda z tych metod wnosi kluczowy wkład w kompleksową strategię diagnostyki COVID-19, pozwalając na lepsze zrozumienie dynamiki pandemii i podejmowanie informowanych decyzji w zakresie zdrowia publicznego.

### Znane algorytmy klasyfikacji

1. UNet + Three D Deep Network

W pracy autorstwa C. Zheng, zastosowano model słabo nadzorowanego uczenia głębokiego, który przetwarzał objętości CT 3D, aby przewidzieć prawdopodobieństwo zakażenia COVID-19. Badanie wykorzystało 499 tomografii CT do treningu i 131 do testowania, osiągając AUC ROC 0,959 i AUC PR 0,976. Model był w stanie szybko przetwarzać dane, wykorzystując dedykowaną jednostkę GPU.

Zalety

* Wysoka dokładność: Wysokie wartości AUC ROC i AUC PR wskazują na dużą dokładność modelu w wykrywaniu COVID-19.
* Efektywność czasowa: Model jest w stanie przetwarzać pojedynczą objętość CT w ciągu zaledwie 1,93 sekundy.
* Wykorzystanie słabo nadzorowanego uczenia: Model nie wymaga szczegółowego oznaczania zmian w CT, co jest znaczącym ułatwieniem w warunkach szybkiego rozprzestrzeniania się epidemii.

Wady

* Ograniczona liczba danych: Badanie opiera się na stosunkowo niewielkiej próbie, co może wpłynąć na generalizację wyników.
* Brak różnorodności danych: Dane pochodzą z jednego szpitala, co może ograniczać ich uniwersalność.
* Brak wyjaśnionej architektury sieci: Jako metoda oparta na głębokim uczeniu, algorytm ma ograniczoną interpretowalność, co może być barierą w klinicznym zastosowaniu oraz dalszym rozwoju algorytmu.
* Możliwe niedoskonałości w segmentacji płuc: Model U-Net użyty do segmentacji płuc nie wykorzystywał informacji czasowych, co mogło wpłynąć na jakość maski płuc.

Badanie demonstruje potencjał wykorzystania głębokiego uczenia w celu szybkiego i dokładnego wykrywania COVID-19 na podstawie tomografii CT klatki piersiowej. Wysoka dokładność i szybkość przetwarzania są obiecujące, choć konieczne są dalsze badania, aby zwiększyć generalizację i rozwiązać problemy interpretowalności. Badanie to jest krokiem naprzód w wykorzystaniu AI do walki z pandemią COVID-19, ale należy je traktować z ostrożnością ze względu na ograniczenia danych i wyzwania związane z "czarną skrzynką" algorytmów uczenia maszynowego.

1. M-Inceoption

W pracy autorstwa S. Wang, opisano rozwój i walidację algorytmu głębokiego uczenia do wykrywania COVID-19 na podstawie obrazów tomografii komputerowej (CT) klatki piersiowej. Badanie obejmowało 1065 obrazów CT, w tym 325 przypadków potwierdzonych COVID-19. Zastosowano zmodyfikowany model Inception w celu klasyfikacji obrazów. Wewnętrzna walidacja modelu wykazała dokładność 89,5%, a zewnętrzna - 79,3%. Badanie pokazuje potencjalną skuteczność głębokiego uczenia w szybkiej i dokładnej diagnostyce COVID-19.

Zalety

* Wysoka dokładność: Model wykazał wysoką dokładność, zwłaszcza w walidacji wewnętrznej.
* Szybka diagnostyka: Algorytm zapewnia szybką diagnostykę, co jest kluczowe w szybkiej reakcji na pandemię.

Wady

* Ograniczenia związane z zewnętrzną walidacją: Dokładność w walidacji zewnętrznej była niższa niż w walidacji wewnętrznej.
* Potrzeba większej liczby danych: Większa ilość danych mogłaby poprawić dokładność i ogólną skuteczność modelu.
* Brak pełnej weryfikacji klinicznej: Model wymaga dalszych badań w realnych warunkach klinicznych.
* Ryzyko nadmiernego dopasowania: Istnieje ryzyko, że model jest nadmiernie dopasowany do danych treningowych.

Badanie pokazuje obiecujące wyniki wykorzystania głębokiego uczenia do diagnostyki COVID-19 na podstawie obrazów CT. Model wykazał wysoką dokładność w walidacji wewnętrznej, ale wymaga dalszych badań, w tym większej liczby danych i weryfikacji klinicznej, aby potwierdzić jego skuteczność i przydatność w praktyce klinicznej. Wnioski te wskazują na potencjalną rolę sztucznej inteligencji w walce z pandemią COVID-19.

1. CCBlock

Dokument prezentuje badanie nad wykorzystaniem głębokiego uczenia do automatycznej diagnozy COVID-19 przy użyciu obrazów rentgenowskich klatki piersiowej. Badanie wykorzystuje zmodyfikowany model VGG (Visual Geometry Group) z dodanym blokiem Convolutional COVID (CCBlock) do diagnozowania i rozróżniania pacjentów z COVID-19, zapaleniem płuc i zdrowych osób. Zastosowano zbiór danych zawierający 1828 zdjęć rentgenowskich, w tym 310 przypadków COVID-19, 864 przypadków zapalenia płuc i 654 zdrowych osób. Wyniki pokazują wysoką skuteczność modelu, osiągając ogólną dokładność 98,52% dla dwóch klas i 95,34% dla trzech klas.

Zalety

* Wysoka dokładność: Model osiągnął bardzo wysoką dokładność w diagnozowaniu COVID-19.
* Różnicowanie stanów zdrowia: Skuteczność modelu w rozróżnianiu COVID-19 od zapalenia płuc i zdrowych osób.
* Szybka diagnoza: Możliwość szybkiego przetwarzania obrazów i diagnozowania.

Wady

* Ograniczona różnorodność danych: Dane pochodzą z publicznie dostępnych platform, co może wpłynąć na uniwersalność modelu.
* Ryzyko nadmiernego dopasowania: Model może być nadmiernie dopasowany do specyficznych cech w danych treningowych.
* Potrzeba weryfikacji klinicznej: Konieczność dalszych badań w środowisku klinicznym, aby potwierdzić skuteczność modelu.
* Zależność od jakości obrazów rentgenowskich: Skuteczność modelu może zależeć od jakości i rodzaju obrazów rentgenowskich używanych do treningu i testowania.

Badanie prezentuje obiecujące wyniki w wykorzystaniu głębokiego uczenia do automatycznej diagnozy COVID-19 na podstawie zdjęć rentgenowskich. Model CCBlock wykazuje wysoką dokładność w rozróżnianiu COVID-19, co może znacząco wspierać proces diagnozy, szczególnie w obliczu wyzwań związanych z pandemią. Jednakże konieczne są dalsze badania, w tym weryfikacja kliniczna, aby potwierdzić skuteczność modelu w praktyce medycznej.

1. DarkCovidNet

Dokument przedstawia rozwój modelu głębokiego uczenia, nazwanego DarkCovidNet, do automatycznej detekcji przypadków COVID-19 za pomocą obrazów rentgenowskich klatki piersiowej. Model został przetestowany na 1125 obrazach (w tym 125 przypadków COVID-19), osiągając dokładność klasyfikacji 98,08% dla dwóch klas (COVID-19 vs. Brak Znalezisk) i 87,02% dla trzech klas (COVID-19 vs. Brak Znalezisk vs. Zapalenie płuc). Autorzy podkreślają potencjał wykorzystania głębokiego uczenia w szybkiej diagnostyce COVID-19, zwracając uwagę na możliwość zastosowania modelu w zdalnych regionach i jako wsparcie dla radiologów.

Zalety

* Wysoka dokładność: Model wykazuje imponującą skuteczność w diagnozowaniu COVID-19.
* Automatyzacja procesu diagnostycznego: Ułatwia proces diagnostyczny, oferując szybką i automatyczną analizę obrazów rentgenowskich.
* Wsparcie dla radiologów: Może służyć jako dodatkowe narzędzie dla specjalistów, zwłaszcza w regionach z ograniczonym dostępem do opieki zdrowotnej.
* Zastosowanie w diagnostyce zdalnej: Model może być użyteczny w środowiskach zdalnych, gdzie brakuje specjalistów.

Wady

* Ograniczona liczba danych COVID-19: Wykorzystano jedynie 125 obrazów potwierdzonych przypadków COVID-19, co może ograniczać generalizację wyników.
* Potrzeba dalszych badań i weryfikacji: Konieczne są dalsze badania kliniczne, aby potwierdzić efektywność modelu w różnorodnych środowiskach medycznych.
* Zależność od jakości obrazowania: Skuteczność modelu może zależeć od jakości obrazów rentgenowskich.
* Ryzyko błędnej interpretacji: Pomimo wysokiej dokładności, istnieje ryzyko błędnej interpretacji w przypadku obrazów o niskiej jakości lub specyficznych przypadków medycznych.

Badanie prezentuje obiecujący model DarkCovidNet, który może znacząco przyczynić się do poprawy procesu diagnozowania COVID-19 poprzez automatyczną analizę obrazów rentgenowskich. Model wykazuje wysoką dokładność i może być szczególnie przydatny w zdalnych lub niedostatecznie zaopatrzonych regionach. Niemniej jednak, konieczne są dalsze badania i weryfikacje kliniczne, aby potwierdzić skuteczność modelu w praktyce i ocenić jego zastosowanie w różnych środowiskach medycznych.

### Podsumowanie znanych algorytmów klasyfikacji

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Architektura** | **Typy obrazów** | **Zbiór danych** | **Zalety** | **Wady** | **Dokładność [%]** |
| UNet + Three D Deep Network | Tomografia komputerowa (CT) | 630 zdjęć  (brak informacji na temat zbalansowania zbioru) | + szybkie przetwarzanie  + wysoka dokładność  + wykorzystanie słabo nadzorowanego uczenia | - ograniczona liczba danych  - dane tylko z jednego szpitala, - brak dokładnego opisu architektury sieci neuronowej | 90.8 |
| M-Inceoption | Tomografia komputerowa (CT) | 1065 (325 COVID-19) | + szybka diagnostyka  + potencjalne zastosowanie kliniczne  + zastosowanie transfer learningu | - ograniczenia związane z zewnętrzną walidacją  - potrzeba większej liczby danych  - niska dokładność w porównaniu z innymi modelami | 85.2 |
| CCBlock | Zdjęcia rentgenowskie | 1828 (310 COVID-19, 864 zapalenie płuc, 654 zdrowe) | + wysoka dokładność  + różnicowanie stanów zdrowia  + szybka diagnoza | - ograniczona różnorodność danych  - ryzyko nadmiernego dopasowania | 98.86 |
| DarkCovidNet | Zdjęcia rentgenowskie | 1125 (125 COVID-19) | + wysoka dokładność | - ograniczona różnorodność danych  - ryzyko nadmiernego dopasowania | 98.08 |

W badaniach dotyczących wykrywania COVID-19 za pomocą technik obrazowania medycznego i algorytmów głębokiego uczenia, przedstawione cztery metody wykazują znaczący potencjał. Metoda opisana przez Zhenga i współautorów wykorzystuje słabo nadzorowane uczenie się do analizy obrazów CT, oferując szybką i skuteczną diagnozę, choć jej dokładność nie została określona. Wang i współautorzy opracowali algorytm na podstawie zmodyfikowanego modelu transferu uczenia Inception dla obrazów CT, osiągając znaczną dokładność, jednak z ograniczeniami związanymi z danymi. Z kolei Al-Bawi i współautorzy prezentują CCBlock, oparty na VGG-16 i CCBlock, dla zdjęć rentgenowskich, osiągając wysoką dokładność w różnicowaniu COVID-19, zapalenia płuc i normalnych wyników. Ostatnia metoda, opracowana przez Ozturka i współautorów, używa DarkCovidNet do analizy zdjęć rentgenowskich, również charakteryzując się wysoką dokładnością i zdolnością do klasyfikacji wieloklasowej. Wszystkie metody, mimo ich zalet, wymagają dalszych badań i walidacji, szczególnie w kontekście zróżnicowanych populacji i jakości danych obrazowych.

## Narzędzia

Przy realizacji pracy dyplomowej, wykorzystano wiele narzędzi, które przyczyniły się do realizacji celów badawczych. Poniżej przedstawione są szczegółowe informacje na temat narzędzi użytych w trakcie opracowywania.

### Sprzęt

Wszystkie testy algorytmu zostały przeprowadzone na komputerze stacjonarnym o następujących parametrach, które są istotne dla metod uczenia maszynowego:

|  |  |
| --- | --- |
| Procesor | AMD Ryzen 5 5600X 6-Core, 4.20 GHz |
| Pamięć RAM | 16 GB |
| System operacyjny | Windows 10 Pro, 64-bitowy |
| Karta graficzna | RTX 3080 (wbudowane 12 GB VRAM) |

### Stos technologiczny

1. Język programowania Python wersja 3.10

Python to wysokopoziomowy język programowania, który jest szeroko stosowany w dziedzinie nauki o danych ze względu na swoją prostotę i potężne biblioteki do analizy danych. W tym badaniu, Python został użyty do implementacji algorytmu klasyfikacji oraz do przetwarzania i analizy danych. Ponad to wykorzystano nastęujące biblioteki:

* Scikit-learn – specjalistyczna biblioteka w języku Python, zapewniająca szeroki wybór narzędzi uczenia maszynowego. Obejmuje ona różne metody klasyfikacji, regresji i grupowania, w tym drzewa decyzyjne, maszyny wektorów nośnych i klasteryzację k-średnich. W tej pracy wykorzystano Scikit-learn do implementacji i ewaluacji modeli klasyfikacji.
* Pandas – biblioteka służąca do manipulacji i analizy danych. Umożliwia ona łatwe wczytywanie, czyszczenie i transformację danych, co jest kluczowe w procesie przygotowania danych do modelowania. W pracy wykorzystano Pandas do wczytania i przetworzenia danych.
* Matplotlib i Seaborn – biblioteki służące do wizualizacji danych. Używane są do tworzenia wykresów i diagramów, które pomagają zrozumieć dane i wyniki analiz. W tej pracy wykorzystano te narzędzia do tworzenia wizualizacji, które pomogły zrozumieć strukturę danych i ocenić wyniki modelowania.
* NumPy – biblioteka, która dodaje wsparcie dla dużych, wielowymiarowych tablic i macierzy, wraz z dużą biblioteką matematycznych funkcji do operacji na tych tablicach. W tej pracy, NumPy zostało użyte do obsługi obliczeń numerycznych.
* PyTorch – otwarte oprogramowanie machine learning rozwijane przez sztuczną inteligencję Facebooka. To narzędzie, które zapewnia maksymalną elastyczność i prędkość podczas implementacji głębokiego uczenia. W pracy wykorzystano PyTorch do budowy i szkolenia głębokich sieci neuronowych

1. CUDA i cuDNN

CUDA to platforma obliczeniowa i model programowania stworzony przez firmę NVIDIA, która umożliwia wykorzystanie mocnych jednostek obliczeniowych GPU (Graphics Processing Units) do przeprowadzania obliczeń ogólnego przeznaczenia. W przypadku tej pracy, CUDA zostało użyte do przyspieszenia obliczeń związanych z uczeniem maszynowym, przede wszystkim w procesie trenowania głębokich sieci neuronowych. cuDNN to biblioteka GPU-accelerowana dla głębokich sieci neuronowych stworzona przez NVIDIA. cuDNN dostarcza wysoce zoptymalizowane funkcje pierwotne do głębokiego uczenia, które są wykorzystywane przez ramy pracy takie jak TensorFlow i PyTorch. W tym badaniu, cuDNN została użyta do zapewnienia wydajnych operacji na poziomie niskiego poziomu, takich jak konwolucje, pooling i normalizacje, które są podstawą dla modeli głębokiego uczenia.

Wykorzystanie CUDA i cuDNN jest kluczowe dla przeprowadzenia obliczeń związanych z uczeniem głębokim na sprzęcie wykorzystującym technologię GPU. Te technologie nie tylko znacząco przyspieszają czas treningu modeli, ale również umożliwiają pracę z dużymi zbiorami danych i skomplikowanymi modelami, które nie byłyby praktyczne lub nawet możliwe do przetworzenia przy użyciu tradycyjnych CPU.

1. Visual Studio Code oraz Jupyter Notebook

Visual Studio Code to edytor kodu źródłowego stworzony przez Microsoft. Jest dostępny na różne platformy systemowe, takie jak Windows, macOS i Linux. VS Code oferuje szeroki zakres funkcji, takich jak wsparcie dla wielu języków programowania, integracja z systemami kontroli wersji, debugowanie, inteligentne podpowiedzi kodu (IntelliSense), a także możliwość rozszerzania funkcjonalności przez dodatki. Jest ceniony za swoją lekkość, wydajność i elastyczność, co sprawia, że jest popularnym wyborem wśród programistów pracujących w różnych technologiach.

Jupyter Notebook to interaktywne środowisko programistyczne, które umożliwia tworzenie i udostępnianie dokumentów zawierających kod, równania, wizualizacje oraz tekst narracyjny. Jest szczególnie popularne w analizie danych, naukach ścisłych i uczeniu maszynowym, umożliwiając łatwe eksperymentowanie z kodem oraz prezentację wyników w czytelny sposób.

### Podstawowe modele i metody uczenia maszynowego

W uczeniu maszynowym wyróżniamy trzy główne rodzaje uczenia: nadzorowane, nienadzorowane i wzmacniane. Każde z nich charakteryzuje się unikalnymi cechami i znajduje zastosowanie w różnych dziedzinach.

**Uczenie Nadzorowane (Supervised Learning):**

Charakterystyka: W uczeniu nadzorowanym model uczy się na podstawie etykietowanych danych, czyli danych, w których każdy przykład jest przypisany do określonej etykiety lub wyniku. Na przykład, w zbiorze danych obrazów, każdy obraz może być oznaczony jako "kot" lub "pies". Model uczy się wzorców i cech z tych danych, które pozwalają mu przewidywać etykiety dla nowych, nieoznakowanych przykładów.

Zastosowanie: Jest szeroko stosowany w problemach klasyfikacji (np. rozpoznawanie obrazów, diagnozowanie chorób na podstawie badań medycznych) i regresji (np. prognozowanie cen nieruchomości, przewidywanie wartości akcji).

**Uczenie Nienadzorowane (Unsupervised Learning):**

Charakterystyka: W uczeniu nienadzorowanym model pracuje na nieetykietowanych danych, co oznacza, że dane wejściowe nie są przypisane do żadnych kategorii, etykiet czy wyników. Zamiast tego, algorytmy próbują znaleźć wzorce, grupowania lub relacje w danych. Metody uczenia nienadzorowanego mogą identyfikować wspólne cechy w danych i grupować je według podobieństwa (klasterowanie) lub redukować wymiarowość danych.

Zastosowanie: Uczenie nienadzorowane znajduje zastosowanie w analizie skupień (np. segmentacja klientów w marketingu), redukcji wymiarowości (np. wizualizacja dużych zbiorów danych), wykrywaniu anomalii (np. identyfikacja oszustw finansowych).

**Uczenie Wzmacniane (Reinforcement Learning):**

Charakterystyka: Uczenie wzmacniane różni się od obu powyższych typów tym, że model (często nazywany agentem) uczy się poprzez interakcje z otoczeniem. Agent podejmuje decyzje (akcje) i otrzymuje informacje zwrotne w postaci nagród lub kar. Celem jest maksymalizacja sumy nagród. Model uczy się, jakie działania prowadzą do największych nagród, często poprzez metodę prób i błędów.

Zastosowanie: Jest stosowane głównie w obszarach wymagających podejmowania decyzji sekwencyjnych, takich jak gry (np. szachy, Go), robotyka (np. automatyczne prowadzenie pojazdów) czy optymalizacja procesów (np. automatyczne zarządzanie systemami logistycznymi).

Każdy z tych rodzajów uczenia ma swoje mocne i słabe strony oraz jest odpowiedni do różnych typów problemów. Wybór odpowiedniego rodzaju uczenia zależy od charakteru danych, celów projektu i specyfiki problemu, który ma być rozwiązany.

1. Drzewa decyzyjne

Są to popularne narzędzia używane w uczeniu maszynowym i analizie danych do klasyfikacji lub regresji. Nazwa "drzewo decyzyjne" pochodzi od sposobu, w jaki algorytm przedstawia decyzje - jako drzewo o wielu gałęziach, które prowadzą do różnych wyników.

Struktura drzewa decyzyjnego:

Drzewo decyzyjne składa się z węzłów decyzyjnych i liści. Każdy węzeł decyzyjny reprezentuje cechę lub atrybut w zestawie danych, gałęzie reprezentują decyzje, a każdy liść reprezentuje wynik. Węzeł na szczycie drzewa, z którego zaczynają się wszystkie inne, nazywany jest węzłem korzenia.

Jak działa drzewo decyzyjne:

Podczas tworzenia drzewa decyzyjnego algorytm zaczyna od węzła korzenia, a następnie dzieli dane na podzbiory, które zawierają możliwie najbardziej jednorodne wartości wynikowe. Wybór atrybutu, który służy do podziału zestawu danych, zależy od statystycznego testu czystości, takiego jak entropia, indeks Gini lub błąd klasyfikacji.

Ten proces jest powtarzany rekurencyjnie, tworząc w ten sposób coraz to nowe węzły decyzyjne aż do osiągnięcia określonego kryterium zatrzymania, na przykład kiedy wszystkie dane w węźle należą do tej samej klasy lub kiedy osiągnięto określoną głębokość drzewa.

Zalety drzew decyzyjnych:

* Drzewa decyzyjne są bardzo intuicyjne i łatwe do zrozumienia, co czyni je atrakcyjnym wyborem, gdy interpretowalność jest ważna. Są one również nieparametryczne, co oznacza, że działają dobrze, nawet jeśli rzeczywiste relacje między cechami a wynikiem są skomplikowane i nie-liniowe.
* Drzewa decyzyjne mogą także radzić sobie z brakującymi danymi oraz są w stanie obsługiwać zarówno dane kategoryczne, jak i numeryczne.

Wady drzew decyzyjnych:

* Jednym z głównych wyzwań związanych z drzewami decyzyjnymi jest tendencja do overfittingu, czyli tworzenia zbyt skomplikowanych modeli, które nie generalizują dobrze na nowych danych. Może to być kontrolowane przez odpowiednie przycinanie drzewa i ustawianie limitów na minimalną liczbę próbek w liściu lub maksymalną głębokość drzewa.
* Innym problemem jest fakt, że drzewa decyzyjne są bardzo czułe na niewielkie zmiany w danych, co może prowadzić do zupełnie innego drzewa. Ten problem jest często łagodzony poprzez użycie wielu drzew decyzyjnych, co prowadzi do takich technik jak lasy losowe.

Mimo tych wyzwań, drzewa decyzyjne są potężnym narzędziem w dziedzinie uczenia maszynowego i mają wiele zastosowań, od medycyny, przez finanse, po inżynierię i nauki ścisłe.

1. Lasy losowe

Jest to technika uczenia maszynowego oparta na drzewach decyzyjnych. Zasada działania lasów losowych polega na generowaniu wielu drzew decyzyjnych podczas procesu treningowego, a następnie na klasyfikacji nowych obserwacji poprzez większościowe głosowanie wyników tych drzew. Ta technika jest bardzo popularna ze względu na swoją skuteczność i prostotę implementacji.

Struktura lasu losowego:

Las losowy, jak sama nazwa sugeruje, składa się z wielu drzew decyzyjnych. Każde drzewo jest tworzone niezależnie od siebie poprzez losowe wybieranie próbek ze zbioru danych treningowych z zastąpieniem (procedura zwana bootstrap), a następnie losowe wybieranie podzbioru cech do podziału węzłów drzewa. Na końcu, wynik klasyfikacji lasu losowego to klasa, która otrzymała najwięcej głosów od wszystkich drzew.

Jak działa las losowy:

Podczas procesu treningu, każde drzewo jest tworzone poprzez losowe wybieranie próbek ze zbioru danych treningowych, a następnie podziału na podzbiory na podstawie losowo wybranego podzbioru cech. Po utworzeniu każdego drzewa, nie jest ono przycinane, co często prowadzi do indywidualnych drzew będących overfitted. Jednak, ze względu na losowość procesu tworzenia lasu, overfitting jednego drzewa jest zwykle zrównoważony przez underfitting innych, co prowadzi do ostatecznego modelu, który dobrze generalizuje na nowych danych.

Zalety lasów losowych:

* Lasy losowe są znane ze swojej zdolności do radzenia sobie z dużą liczbą cech i odporności na overfitting, dzięki czemu są szczególnie skuteczne w sytuacjach, gdzie mamy do czynienia z dużymi i złożonymi zestawami danych.
* Są także w stanie efektywnie radzić sobie z brakującymi danymi i są mniej wrażliwe na outlierów niż wiele innych metod uczenia maszynowego.

Wady lasów losowych:

* Pomimo powyższych zalet, lasy losowe nie są pozbawione wad. Jednym z najważniejszych ograniczeń jest to, że są one trudniejsze do interpretacji niż pojedyncze drzewa decyzyjne. Ponadto, choć są one zazwyczaj odporne na overfitting, mogą się przetrenować, jeśli liczba drzew jest zbyt duża.

Pomimo tych wyzwań, lasy losowe są potężnym narzędziem w dziedzinie uczenia maszynowego i mają szerokie zastosowanie w różnych dziedzinach, takich jak medycyna, finanse czy nauki ścisłe.

1. K najbliższych sąsiadów (K-Nearest Neighbors, K-NN)

To jedna z najprostszych metod uczenia maszynowego. Algorytm K-NN jest metodą opartą na instancjach, co oznacza, że nie tworzy on eksplicytnego modelu podczas treningu, lecz zamiast tego przechowuje wszystkie dane treningowe. Następnie, podczas fazy predykcji, K-NN klasyfikuje nowe obserwacje na podstawie ich podobieństwa do przechowywanych przykładów.

Jak działa K najbliższych sąsiadów:

Główna idea metody K-NN polega na tym, że obserwacje są podobne do siebie i że podobne obserwacje mają podobne wyniki. Dla danego, nieklasyfikowanego jeszcze przykładu, algorytm K-NN szuka 'k' najbliższych sąsiadów w przestrzeni cech treningowych. Następnie, na podstawie klas tych sąsiadów, nowy przykład jest klasyfikowany przez większościowe głosowanie - najczęściej występująca klasa wśród 'k' najbliższych sąsiadów jest przypisywana jako klasa dla nowego przykładu.

Zalety K najbliższych sąsiadów:

* K-NN jest bardzo prostym algorytmem, który jest łatwy do zrozumienia i implementacji. Nie wymaga on żadnego treningu, co oznacza, że nowe dane mogą być dodawane w dowolnym momencie, bez konieczności przeliczania modelu. K-NN potrafi dobrze radzić sobie z problemami, które są nieliniowe, co jest dużym atutem, szczególnie w kontekście danych, które są trudne do modelowania za pomocą technik liniowych.

Wady K najbliższych sąsiadów:

* Jedną z głównych wad K-NN jest fakt, że może być on bardzo kosztowny pod względem obliczeniowym i pamięciowym, zwłaszcza dla dużych zbiorów danych, ponieważ przechowuje on wszystkie dane treningowe. Ponadto, K-NN jest wrażliwy na skalę cech, co oznacza, że cechy muszą być normalizowane, aby algorytm działał poprawnie. K-NN może także mieć trudności z klasyfikacją, kiedy dane są nierównomiernie rozłożone między klasy.

Mimo tych wyzwań, K najbliższych sąsiadów jest prostym, ale skutecznym narzędziem w dziedzinie uczenia maszynowego i ma szerokie zastosowanie w różnych dziedzinach, takich jak medycyna, systemy rekomendacji, rozpoznawanie wzorców i wiele innych.

1. Uczenie głębokie (Deep Learning)

Uczenie głębokie stało się kluczowym narzędziem w dziedzinie uczenia maszynowego i analizy danych, szczególnie w przypadkach, gdzie kluczowe jest modelowanie skomplikowanych wzorców i relacji w dużych zbiorach danych. Jego nazwa pochodzi od struktury sieci neuronowych, które składają się z wielu, często setek, "głębokich" warstw.

Struktura uczenia głębokiego:

Uczenie głębokie polega na wykorzystaniu szeregów hierarchicznych warstw sztucznych neuronów, zwanych sieciami neuronowymi. Każda warstwa w tej sieci przetwarza informacje na wyższym poziomie abstrakcji niż poprzednia. To oznacza, że podczas gdy dolne warstwy mogą nauczyć się rozpoznawać kształty lub krawędzie, wyższe warstwy mogą komponować te elementy w bardziej skomplikowane struktury, takie jak twarze czy obiekty.

Jak działa uczenie głębokie:

Podczas procesu uczenia, sieć neuronowa optymalizuje swój model danych, dostosowując wagi połączeń między neuronami na podstawie błędu pomiędzy predykcją sieci a rzeczywistymi danymi. Ten proces optymalizacji zwykle odbywa się za pomocą algorytmu zwanej propagacji wstecznej i może wymagać dużej ilości danych i mocy obliczeniowej.

Zalety uczenia głębokiego:

* Jednym z największych atutów uczenia głębokiego jest jego zdolność do automatycznego uczenia się cech z danych, co eliminuje potrzebę ręcznego projektowania cech. Jest to szczególnie przydatne w zadaniach związanych z obrazem, dźwiękiem czy tekstem, gdzie ręczne projektowanie cech może być trudne i pracochłonne.
* Ponadto, ze względu na swoją złożoność, uczenie głębokie jest w stanie modelować skomplikowane zależności w danych, co czyni je skutecznym narzędziem w wielu dziedzinach, od przetwarzania języka naturalnego po samochody autonomiczne.

Wady uczenia głębokiego:

* Jednym z głównych wyzwań związanych z uczeniem głębokim jest jego wymóg dotyczący dużej ilości danych i zasobów obliczeniowych. Bez odpowiedniej ilości danych, modele uczenia głębokiego mogą łatwo ulec overfittingowi, co oznacza, że mogą zbyt skomplikować model i nie generalizować dobrze na nowych danych.
* Podobnie jak drzewa decyzyjne, uczenie głębokie jest wrażliwe na zmiany w danych wejściowych, ale istnieją techniki, takie jak augmentacja danych czy normalizacja wsadowa, które mogą pomóc złagodzić ten problem.

Pomimo tych wyzwań, uczenie głębokie stało się podstawowym narzędziem w dziedzinie sztucznej inteligencji i ma wiele zastosowań, od rozpoznawania obrazów, przez syntezę mowy, po rekomendacje produktów.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Metoda** | **Struktura** | **Działanie** | **Zalety** | **Wady** |
| Drzewa decyzyjne | Węzły decyzyjne i liście | Dzielenie danych na podzbiory na podstawie testów czystości | Intuicyjne, nieparametryczne, radzą sobie z brakującymi danymi | Podatność na overfitting, wrażliwość na zmiany w danych |
| Lasy losowe | Wiele drzew decyzyjnych | Generowanie wielu drzew i klasyfikacja przez większościowe głosowanie | Odporność na overfitting, skuteczne przy dużych zestawach danych | Trudne do interpretacji, możliwość przetrenowania przy zbyt dużej liczbie drzew |
| K-NN | Brak eksplicytnego modelu, przechowuje wszystkie dane | Klasyfikacja na podstawie 'k' najbliższych sąsiadów (podobnych wyników) | Proste, nie wymaga treningu, efektywne w nieliniowych problemach | Kosztowne obliczeniowo, wrażliwe na skalę cech |
| Uczenie głębokie | Warstwy sztucznych neuronów | Optymalizacja modelu poprzez dostosowywanie wag | Automatyczne uczenie się cech, skuteczne w modelowaniu złożonych zależności | Wymaga dużych ilości danych i zasobów obliczeniowych, podatność na overfitting bez danych |

Podsumowując, omówione zostały różne techniki, w tym drzewa decyzyjne, lasy losowe, K najbliższych sąsiadów (K-NN) i uczenie głębokie. Z zebranych informacji wynika, że najbardziej obiecującą metodą do zastosowania w algorytmach klasyfikacji wydaje się być uczenie głębokie. Jego główne zalety to zdolność do automatycznego wykrywania i uczenia się skomplikowanych wzorców w dużych zbiorach danych, co jest kluczowe w wielu zaawansowanych zastosowaniach, takich jak przetwarzanie języka naturalnego, analiza obrazów czy rozpoznawanie wzorców. Uczenie głębokie oferuje również większą elastyczność i adaptacyjność do różnorodnych typów danych w porównaniu do innych metod.

### Zbiór danych

Odpowiedni zbiór danych jest kluczowy przy tworzeniu algorytmu klasyfikacji zdjęć rentgenowskich klatki piersiowej w diagnozowaniu infekcji COVID-19. Jakość, reprezentatywność i różnorodność danych bezpośrednio wpływają na skuteczność i dokładność algorytmu. Dla modelu głębokiego uczenia, który ma identyfikować cechy COVID-19 na zdjęciach RTG, istotne jest, aby dane były zróżnicowane pod względem płci, wieku, grup etnicznych i innych czynników klinicznych. Brak różnorodności w tych aspektach może prowadzić do stronniczości modelu, na przykład, jeśli dane są przeważnie z jednej grupy etnicznej lub płci, model może być mniej dokładny w diagnozowaniu osób z innych grup. Takie braki w różnorodności danych mogą prowadzić do nieodpowiednich diagnoz, szczególnie w przypadku mniejszości etnicznych lub płci, które nie są adekwatnie reprezentowane w danych treningowych. Ponadto, zdjęcia RTG mogą wykazywać różnice w zależności od technik obrazowania i warunków klinicznych, co także powinno być uwzględnione w zbiorze danych. Oddzielne zbiory testowe i walidacyjne, odzwierciedlające różnorodność rzeczywistego świata, są niezbędne do oceny, jak dobrze model radzi sobie z diagnozowaniem COVID-19 w różnych grupach ludności. W rezultacie, starannie skomponowany i różnorodny zbiór danych jest niezbędny, aby algorytm klasyfikacji RTG w kontekście COVID-19 był skuteczny, sprawiedliwy i wiarygodny w różnorodnych środowiskach klinicznych, minimalizując ryzyko błędnych diagnoz i zapewniając równomierną jakość opieki dla wszystkich pacjentów.

Idealny zbiór danych do uczenia głębokiego powinien posiadać następujące cechy:

* Duża objętość: Uczenie głębokie wymaga dużych zbiorów danych do skutecznego trenowania modeli, co pozwala na uchwycenie złożonych wzorców.
* Różnorodność: Zbiór danych powinien być reprezentatywny dla problemu, który ma być rozwiązany, i zawierać różnorodne przypadki, aby model mógł nauczyć się generalizować.
* Wysoka Jakość i czystość Danych: Dane powinny być wolne od błędów i nieścisłości, co zapewnia lepszą dokładność modelu.
* Zbalansowanie: W przypadku problemów klasyfikacyjnych, idealnie jest, gdy poszczególne klasy są równomiernie reprezentowane w danych.
* Oznakowanie/Annotacje: W przypadku uczenia nadzorowanego, dane powinny być właściwie oznakowane, co umożliwia modelowi uczenie się na podstawie przykładów.
* Zgodność Etyczna i Prawna: Zbiór danych powinien być zgodny z przepisami dotyczącymi prywatności i etyki, szczególnie w przypadku wrażliwych danych osobowych.

Do stworzenia algorytmu wykorzystano dwa zbiory danych:

1. „COVIDGR”, który został opisany w artykule "COVIDGR Dataset and COVID-SDNet Methodology for Predicting COVID-19 Based on Chest X-Ray Images" i jest specjalnie przygotowanym zbiorem danych do badania i detekcji COVID-19 na podstawie zdjęć rentgenowskich klatki piersiowej. Został stworzony we współpracy z Hospital Universitario Clínico San Cecilio w Granadzie w Hiszpanii. Zbiór zawiera zdjęcia rentgenowskie przedstawiające wszystkie poziomy nasilenia COVID-19, od normalnych przypadków z pozytywnym wynikiem RT-PCR, przez łagodne i umiarkowane, aż po ciężkie przypadki. Zawiera on 426 pozytywnych i 426 negatywnych widoków PA (PosteroAnterior) CX, co jest stosunkowo małą liczbą w porównaniu z wymaganiami modeli uczenia głębokiego, jednak w rzeczywistych warunkach, nie można liczyć na wielkie zbiory danych. COVIDGR obejmuje szeroki zakres przypadków COVID-19, od łagodnych po ciężkie, co zapewnia reprezentatywność zbioru dla różnych manifestacji choroby. Jest to istotne dla modeli, które mają generalizować wyniki na różne typy przypadków. Co więcej, zbiór danych został zebrany we współpracy z placówką medyczną, co sugeruje wysoką jakość i wiarygodność danych. COVIDGR jest zbiorem zbalansowanym, z równą liczbą przypadków pozytywnych i negatywnych, co jest korzystne dla uniknięcia stronniczości modelu i zapewnienia równomiernego treningu dla obu klas. W przeciwnym wypadku model mógłby sztucznie zawyżać swoją dokładność, poprzez częstsze typowanie jednej z klas.Rentgenogramy dla ułatwienia zostały podzielone na dwa foldery, jeden Podsumowując, zbiór danych COVIDGR wydaje się być wartościowym zasobem do badań nad COVID-19, zwłaszcza ze względu na jego różnorodność i zbalansowanie.
2. Drugim zbiorem danych, który jest wykorzystywany do wzbogacenia pierwszego zbioru, jest zbiór dostępny na platformie Mendeley. Zbiór ten zawiera zdjęcia rentgenowskie klatki piersiowej z przypadkami pneumonii, ale z negatywnym wynikiem testu na COVID-19. Integracja tego zbioru danych ma kluczowe znaczenie, aby uniknąć sytuacji, w której model klasyfikacji mógłby błędnie nauczyć się, że jakiekolwiek odbiegnięcie od normy w obrazie RTG klatki piersiowej oznacza obecność COVID-19. Poprzez dodanie zdjęć przedstawiających inne choroby płuc, takie jak pneumonia, model może lepiej rozróżniać między COVID-19 a innymi schorzeniami, co jest istotne dla precyzyjnego diagnozowania. Zbiór danych z Mendeley zawiera obrazy pacjentów z różnymi stopniami zaawansowania pneumonii, co umożliwia modelowi naukę identyfikacji i różnicowania wielu stanów zdrowotnych. To podejście wspiera nie tylko skuteczność algorytmu w diagnozowaniu COVID-19, ale także wzmacnia jego zdolność do generalizacji i prawidłowej klasyfikacji innych chorób płuc, co jest niezbędne w precyzyjnej medycynie i unikaniu błędnych diagnoz.

Podsumowując, kluczowym aspektem tworzenia efektywnego algorytmu klasyfikacji zdjęć rentgenowskich klatki piersiowej dla diagnozowania infekcji COVID-19 jest zgromadzenie odpowiedniego zbioru danych. Jego jakość, reprezentatywność i różnorodność bezpośrednio wpływają na skuteczność i dokładność algorytmu. Istotne jest, aby dane były zróżnicowane, uwzględniając płcie, wiek, grupy etniczne i inne czynniki kliniczne, aby uniknąć stronniczości modelu.

W tym kontekście, zbiór danych „COVIDGR” stanowi cenny zasób, oferując szeroki zakres przypadków COVID-19, od łagodnych po ciężkie, z równą reprezentacją wyników pozytywnych i negatywnych. Jego współtworzenie z placówką medyczną zapewnia wysoką jakość i wiarygodność danych. Niemniej jednak, z uwagi na ograniczoną liczbę przypadków w „COVIDGR”, zbiór danych z Mendeley został dodany do projektu, aby zapewnić dodatkowe przykłady pneumonii z negatywnym wynikiem testu na COVID-19. Pozwala to na bardziej zrównoważone i kompleksowe szkolenie modelu, co jest niezbędne do uniknięcia błędnych diagnoz i zapewnienia równomierniej jakości opieki dla wszystkich pacjentów.

W efekcie, starannie skomponowany i różnorodny zbiór danych jest kluczowy dla skuteczności, sprawiedliwości i wiarygodności algorytmu klasyfikacji RTG w kontekście COVID-19, minimalizując ryzyko błędnych diagnoz. Idealny zbiór danych powinien charakteryzować się dużą objętością, różnorodnością, wysoką jakością i czystością danych, zbalansowaniem oraz właściwym oznakowaniem i annotacjami, z zachowaniem zgodności etycznej i prawnej, szczególnie w przypadku wrażliwych danych osobowych.

## Cel pracy

W niniejszej pracy dyplomowej głównym celem jest opracowanie nowego algorytmu klasyfikacji obrazów rentgenowskich klatki piersiowej, który będzie używany do diagnozowania COVID-19. Intencją jest, aby wspierał on personel medyczny w efektywnym i precyzyjnym rozpoznawaniu przypadków infekcji. Algorytm zostanie dokładnie zbadany i porównany z istniejącymi metodami klasyfikacji oraz innymi dostępnymi technikami diagnostycznymi COVID-19, takimi jak testy RT-PCR i badania serologiczne.

Praca dyplomowa rozpocznie się od przeprowadzenia badań literaturowych, które posłużą do zebrania informacji na temat specyficznych zmian radiologicznych u pacjentów z COVID-19 oraz obecnych metod diagnostycznych. Na podstawie zgromadzonych danych przeprowadzona zostanie analiza obrazów rentgenowskich, co pozwoli na identyfikację typowych wzorców związanych z infekcją.

Następnie skoncentrowano się na tworzeniu algorytmu klasyfikacji, który będzie stosowany weryfikacji, by ocenić jego efektywność i dokładność. W zakończeniu pracy przedstawione zostaną wyniki badań, oceniona zostanie użyteczność opracowanego algorytmu, a także jego potencjalne zastosowanie w praktyce medycznej i możliwości dalszego rozwoju.

Rozwój takiego algorytmu ma potencjał znaczącego wpływu na poprawę jakości diagnostyki radiologicznej w przypadku podejrzenia infekcji COVID-19, co może przyczynić się do szybszej i bardziej skutecznej ścieżki leczenia dla pacjentów. Opracowanie tego algorytmu może również stanowić ważny wkład w dziedzinę medycyny, w szczególności w zakresie zastosowania sztucznej inteligencji w diagnostyce obrazowej.

## Zakres prac

Zakres prac dyplomowych dotyczących opracowania jednego algorytmu klasyfikacji rentgenogramów klatki piersiowej w celu diagnozowania COVID-19 obejmuje następujące etapy:

1. Badania literaturowe: Przegląd aktualnych publikacji naukowych dotyczących technik klasyfikacji obrazów medycznych, w tym rentgenogramów płuc. Skoncentrowano się na analizie metod wykorzystywanych do tej pory w detekcji COVID-19 oraz ocenie potencjału zastosowania sztucznej inteligencji w radiologii.
2. Zbieranie i przygotowanie danych: Kompletowanie odpowiedniego zbioru danych zawierającego obrazy rentgenowskie pacjentów z potwierdzonym i wykluczonym zakażeniem SARS-CoV-2. Dane te są następnie poddawane obróbce wstępnej, w tym segmentacji i anotacji, co jest niezbędne do dalszej analizy i procesu uczenia.
3. Opracowanie algorytmu klasyfikacji: Na podstawie zgromadzonej wiedzy zostanie wybrana i zaimplementowana jedna wybrana metoda klasyfikacji, która będzie porównana z obecnymi rozwiązaniami oraz innymi metodami diagnostycznymi, takimi jak testy RT-PCR i badania serologiczne.
4. Trenowanie i walidacja modelu: Opracowany algorytm zostanie przetestowany na zebranych danych, aby ocenić jego efektywność i dokładność w identyfikacji przypadków COVID-19 na rentgenogramach.
5. Analiza porównawcza i ewaluacja: Przeprowadzona zostanie szczegółowa analiza wyników, w tym porównanie z wynikami innych metod klasyfikacji i standardowych testów diagnostycznych, aby określić przewagi i ograniczenia zaproponowanego rozwiązania.
6. Wnioski i rekomendacje: Na koniec przedstawione zostaną wnioski z badań oraz zalecenia dotyczące potencjalnego zastosowania algorytmu w medycynie klinicznej, a także możliwości jego dalszego rozwoju.

Zakres ten ma na celu nie tylko rozwój nowego narzędzia diagnostycznego, ale także przyczynienie się do lepszego zrozumienia roli i miejsca algorytmów klasyfikacyjnych w kontekście obecnych metod diagnozowania COVID-19.

# Projekt algorytmu klasyfikacji

W niniejszym dziale zostanie przedstawione szczegółowe działanie opracowanego algorytmu klasyfikacji rentgenogramów w kontekście diagnozowania infekcji COVID-19. Zostaną omówione kluczowe aspekty architektury algorytmu, jego funkcjonalności oraz wybrane metody i technologie, które zostały zastosowane w celu efektywnego i precyzyjnego przetwarzania danych obrazowych.

Celem tego działu jest nie tylko zaprezentowanie samego algorytmu, ale także wyjaśnienie i uzasadnienie podjętych decyzji dotyczących konkretnych rozwiązań i technik wykorzystanych w jego konstrukcji. Skupimy się na przedstawieniu, w jaki sposób poszczególne wymagania funkcjonalne i niefunkcjonalne zostały zaadresowane w procesie projektowania i implementacji algorytmu. Wyjaśnione zostaną również kroki podejmowane w procesie optymalizacji modelu, aby zapewnić jego wysoką dokładność i efektywność.

## Analiza wymagań funkcjonalnych i niefunkcjonalnych

Algorytm klasyfikacji rentgenogramów w infekcji COVID-19, opracowany w ramach pracy dyplomowej, powinien spełniać następujące wymagania funkcjonalne i niefunkcjonalne.

### 2.1.1. Wymagania funkcjonalne

* Klasyfikacja obrazów: Algorytm powinien być w stanie skutecznie klasyfikować obrazy rentgenowskie pacjentów na podstawie charakterystycznych cech związanych z infekcją COVID-19.
* Przyjmowanie danych wejściowych: Algorytm powinien być w stanie przyjmować jako dane wejściowe obrazy rentgenowskie w odpowiednim formacie
* Zwracanie wyników: Algorytm powinien zwracać wyniki klasyfikacji, takie jak prawdopodobieństwo infekcji COVID-19 lub binarne etykiety klasyfikacyjne (COVID-19/nie COVID-19).
* Uczenie z nadzorem: Algorytm powinien korzystać z danych etykietowanych do nauki, umożliwiając proces uczenia z nadzorem.
* Optymalizacja modelu: Algorytm powinien umożliwiać dostosowanie i optymalizację parametrów modelu w celu zwiększenia skuteczności klasyfikacji.

### Wymagania niefunkcjonalne

* Dokładność: Algorytm powinien mieć wysoką dokładność klasyfikacji, minimalizując błędy typu I (fałszywie pozytywne) i II (fałszywie negatywne).
* Skalowalność: Algorytm powinien być w stanie przetwarzać duże zbiory danych, skalując się efektywnie wraz ze wzrostem rozmiaru danych.
* Wydajność: Algorytm powinien działać efektywnie, minimalizując czas potrzebny na przetwarzanie danych i klasyfikację obrazów.
* Bezpieczeństwo danych: Wszystkie dane pacjentów przetwarzane przez algorytm powinny być chronione, zgodnie z przepisami o ochronie danych osobowych, takimi jak RODO.
* Zgodność z etyką medyczną: Algorytm powinien być opracowany i stosowany w sposób zgodny z etyką medyczną, co obejmuje uzyskanie odpowiednich zgód na wykorzystanie danych pacjentów do celów badawczych.
* Odtwarzalność: Algorytm powinien być zaprojektowany w taki sposób, aby inne osoby mogły go odtworzyć i zastosować na podstawie dostarczonych informacji, co promuje przejrzystość i reprodukowalność naukową.

## Architektura algorytmu

### Schemat ogólny

Zdecydowano, że uczenie nadzorowane będzie najlepszym podejściem do problemu klasyfikacji, ponieważ opiera się na wykorzystaniu etykietowanych danych, co jest kluczowe w medycznych zastosowaniach obrazowania. Dane zostały podzielone na dwa foldery, N i P, zawierające odpowiednio rentgenogramy z negatywnymi i pozytywnymi diagnozami infekcji COVID-19, co umożliwia modelowi nauczanie się rozpoznawania specyficznych cech związanych z infekcją. Uczenie nadzorowane jest idealne do takich zadań klasyfikacyjnych, gdzie dokładność i zdolność do precyzyjnego rozróżniania między różnymi stanami zdrowia są krytyczne. Dodatkowo, modele nadzorowane często zapewniają wyniki, które można łatwiej interpretować i weryfikować przez specjalistów medycznych, co jest niezbędne w zastosowaniach klinicznych, gdzie błędna diagnoza może mieć poważne konsekwencje.

W ramach algorytmu klasyfikacji rentgenogramów klatki piersiowej dla diagnozowania infekcji COVID-19, zdecydowano się na zastosowanie głębokiej sieci konwolucyjnej (CNN), co jest decyzją podyktowaną wieloma kluczowymi aspektami. Po pierwsze, sieci konwolucyjne specjalizują się w przetwarzaniu danych obrazowych, co jest istotne, ponieważ potrafią one efektywnie identyfikować cechy w obrazach, takie jak kształty, krawędzie i wzory – elementy krytyczne w analizie obrazów medycznych, gdzie detale takie jak wzory tkankowe mogą wskazywać na obecność choroby. Dodatkowo, CNN zachowują przestrzenną integralność obrazu, co oznacza, że mogą one skutecznie analizować i uczyć się przestrzennych relacji między elementami na obrazie – co jest niezbędne, gdyż lokalizacja i struktura zmian w obrazie rentgenowskim mogą dostarczać istotnych informacji diagnostycznych. Kolejną zaletą jest zdolność CNN do redukcji złożoności obrazów poprzez hierarchiczne wyodrębnianie cech, co eliminuje potrzebę ręcznego wyodrębniania cech i zwiększa efektywność całego procesu. Ponadto, CNN są znane ze swojej efektywności w pracy z dużymi i złożonymi zbiorami danych, co jest istotne w medycynie, gdzie dostępne są duże zestawy danych obrazowych. Wreszcie, możliwość stosowania techniki transfer learning w przypadku CNN jest szczególnie cenna w medycynie, umożliwiając wykorzystanie modelu wytrenowanego na jednym zbiorze danych do podobnego zadania na innym zbiorze danych, co jest kluczowe w sytuacjach, gdzie gromadzenie dużych zbiorów etykietowanych danych jest wyzwaniem. Wszystkie te cechy czynią CNN szczególnie odpowiednimi do zastosowań w diagnostyce medycznej, zwłaszcza w analizie obrazów rentgenowskich klatki piersiowej w kontekście diagnozowania infekcji COVID-19.

### Podział danych (cross-walidacja)

W ramach opracowywanego algorytmu klasyfikacji rentgenogramów klatki piersiowej dla diagnozowania infekcji COVID-19, zdecydowano się na podział danych na dwa główne segmenty: 80% danych przeznaczono na trening modelu, a pozostałe 20% stanowi zbiór testowy, wykorzystywany do końcowej oceny wydajności modelu. Kluczowym aspektem tego podejścia jest zastosowanie walidacji krzyżowej na zbiorze treningowym, co jest realizowane poprzez dodatkowy podział tych 80% danych na 5 równych części, zwanych foldami. Takie rozwiązanie pozwala na skuteczniejsze wykorzystanie dostępnych danych treningowych i zwiększa wiarygodność oceny modelu.

Wykorzystanie walidacji krzyżowej, w szczególności 5-krotnej walidacji krzyżowej, ma wiele zalet. Po pierwsze, każda część danych jest wykorzystywana zarówno do treningu, jak i do walidacji, co jest szczególnie ważne w przypadkach, gdy ilość dostępnych danych jest ograniczona. Dzięki temu, model jest w stanie lepiej nauczyć się ogólnych wzorców, a nie tylko specyfik danych treningowych. Po drugie, walidacja krzyżowa pozwala na dokładniejszą ocenę modelu, ponieważ wyniki są uśredniane z wielu iteracji, co redukuje ryzyko przypadkowego błędu. W ten sposób, wyniki są bardziej stabilne i mniej podatne na anomalie specyficzne dla jednego zestawu danych.

Podział 80% danych na trening i 20% na testowanie jest często stosowaną praktyką w uczeniu maszynowym. Pozwala on na zachowanie wystarczającej ilości danych do treningu modelu, jednocześnie zapewniając, że pozostaje wystarczająco duża ilość niezależnych danych do sprawdzenia jego wydajności. Zastosowanie 5 foldów w walidacji krzyżowej jest dobrym kompromisem między dokładnością oceny modelu a złożonością obliczeniową procesu. Każdy z foldów służy jako zbiór walidacyjny dokładnie jeden raz, co zapewnia równomierną reprezentację wszystkich części zbioru treningowego w procesie walidacji. To podejście jest szczególnie korzystne w medycynie, gdzie jakość i dokładność diagnozy są krytyczne, a jednocześnie często mamy do czynienia z ograniczonymi ilościami danych.

### Segmentacja

Segmentacja płuc w rentgenogramach klatki piersiowej odgrywa kluczową rolę w procesie tworzenia algorytmu klasyfikacji dla diagnozowania infekcji COVID-19. Ta technika, opierająca się na zaawansowanych metodach przetwarzania obrazu, w tym technikach uczenia maszynowego i głębokiego uczenia, skupia się na precyzyjnym wyodrębnieniu obszaru płuc z całego obrazu rentgenowskiego. Dzięki temu, możliwe jest skoncentrowanie analizy algorytmu wyłącznie na tych obszarach, które są najbardziej istotne dla diagnozy. Zastosowanie segmentacji płuc ma wiele istotnych korzyści: po pierwsze, pozwala ona na skupienie uwagi algorytmu na kluczowych dla diagnozy obszarach, eliminując tym samym nieistotne dla diagnozy elementy obrazu. Pozwala to zwiększyć skuteczność algorytmu w detekcji specyficznych dla COVID-19 zmian, takich jak zacienienia czy zmiany włókniste. Po drugie, segmentacja znacząco poprawia dokładność algorytmu, ograniczając ryzyko błędnej interpretacji elementów obrazu niezwiązanych z chorobą, co jest szczególnie ważne w kontekście medycznym, gdzie każda pomyłka może mieć poważne konsekwencje. Po trzecie, redukuje zakłócenia i szumy pochodzące od innych struktur anatomicznych, takich jak kości i tkanki miękkie, co jest niezbędne dla czystości sygnału diagnostycznego. Po czwarte, skupienie analizy na obszarze płuc zmniejsza objętość danych do przetwarzania, co przekłada się na zwiększoną efektywność obliczeniową algorytmu, co jest kluczowe przy dużych zbiorach danych. Wreszcie, ułatwia stosowanie różnych technik augmentacji danych i innych procesów przetwarzania, które są spójne i skupione na obszarze zainteresowania, co prowadzi do lepszych wyników treningu. Tak więc, segmentacja płuc nie tylko zwiększa precyzję i skuteczność algorytmu klasyfikacji, ale również poprawia ogólną efektywność i spójność procesu przetwarzania danych, co jest niezmiernie ważne w kontekście diagnozowania infekcji COVID-19.

### Augmentacja

Augmentacja danych to proces polegający na sztucznym zwiększaniu rozmiaru i różnorodności zestawu danych poprzez wprowadzanie modyfikacji do istniejących próbek. W kontekście obrazów, może to obejmować zmiany takie jak obracanie, skalowanie, zmiany perspektywy, dodawanie szumu, zmiany kontrastu czy jasności. Głównym celem augmentacji jest stworzenie bardziej wszechstronnego i zróżnicowanego zestawu danych, który może pomóc modelowi uczenia maszynowego lepiej generalizować i funkcjonować efektywnie w różnych warunkach.

Zalety augmentacji danych:

* Zwiększenie Rozmiaru Danych: Augmentacja pozwala na powiększenie istniejącego zestawu danych, co jest szczególnie przydatne, gdy oryginalny zestaw danych jest mały.
* Zapobieganie Nadmiernemu Dopasowaniu (Overfitting): Poprzez zwiększenie różnorodności danych, augmentacja pomaga zapobiegać sytuacji, w której model zbyt mocno dopasowuje się do danych treningowych i nie radzi sobie dobrze z nowymi, nieznajomymi danymi.
* Poprawa Generalizacji: Model trenowany na zróżnicowanych danych ma większą szansę na skuteczną pracę w różnorodnych warunkach rzeczywistych.
* Symulacja Rzeczywistych Warunków: W medycynie, na przykład, augmentacja może symulować różne warunki obrazowania, co jest istotne dla przygotowania modelu do pracy w realistycznych scenariuszach klinicznych.

Wady augmentacji danych:

* Potencjalne Wprowadzenie Błędów: Nieprawidłowo wykonana augmentacja może wprowadzić nienaturalne artefakty do obrazów, co może prowadzić do błędnych nauk modelu.
* Złożoność Obliczeniowa: Augmentacja danych może zwiększyć wymagania obliczeniowe, co oznacza potrzebę większej mocy obliczeniowej i dłuższego czasu treningu.
* Ryzyko nadmiernego uogólnienia: Jeśli augmentacja jest przesadzona, model może stać się zbyt odporny na zmiany w danych i tracić zdolność do wykrywania subtelnych cech.

W kontekście algorytmu klasyfikacji, augmentacja danych może znacząco wpłynąć na jego skuteczność. Poprawia to zdolność algorytmu do generalizacji, zwiększa jego odporność na różnorodne warunki danych wejściowych i pomaga w uzyskaniu bardziej wiarygodnych wyników. Jednakże, należy zachować równowagę i nie przesadzać z ilością i rodzajem stosowanych technik augmentacji, aby uniknąć wprowadzenia błędów i niepotrzebnej złożoności. Właściwie stosowana augmentacja danych może być kluczowym elementem w tworzeniu wydajnych i dokładnych modeli klasyfikacyjnych, zwłaszcza w dziedzinach, gdzie dane są ograniczone lub bardzo zróżnicowane, jak w medycynie.

### Przygotowanie danych

Przygotowanie danych jest kluczowym etapem w procesie ucznia sieci neuronowej, a jego główne elementy to: segregacja danych w odpowiednią strukturę folderów, zmiana rozmiaru obrazów, konwersja palety barw do monochromatyczności. Przede wszystkim, zmiana rozmiaru zdjęć jest konieczna, ponieważ sieci neuronowe wymagają danych wejściowych o jednolitym rozmiarze. Dostosowanie rozmiaru obrazów zapewnia, że wszystkie próbki będą miały jednakowe wymiary, co jest niezbędne dla efektywnego i spójnego przetwarzania przez model. Konwersja do monochromatyczności (obrazy w skali szarości) pomaga w redukcji złożoności obrazów, eliminując kolory, które często nie niosą istotnych informacji diagnostycznych w kontekście rentgenogramów płucnych. Takie uproszczenie może przyczynić się do szybszego przetwarzania i mniejszych wymagań obliczeniowych, co jest ważne przy dużych zbiorach danych. Ponadto, skupienie się na odcieniach szarości pomaga modelowi skoncentrować na kontrastach i strukturach anatomicznych, które są kluczowe w diagnozie. Mimo, że większość rentgenogramów jest w skali monochromatycznej, to warto mieć pewność, że tak faktycznie jest. Posegregowanie danych w odpowiednią strukturę folderów, zwykle z podziałem na foldery odpowiadające klasom lub kategoriom (np. 'COVID-19' i 'Nie-COVID-19'), jest niezbędne dla organizacji procesu uczenia. Taki podział ułatwia zarządzanie danymi, automatyzację procesu ładowania danych do modelu oraz zapewnia czytelność i łatwość w dostępie do danych. Umożliwia to również łatwą implementację procedur takich jak walidacja krzyżowa i augmentacja danych. W rezultacie, odpowiednie przygotowanie danych, jest niezbędne do stworzenia wydajnego i skutecznego procesu trenowania sieci neuronowej, co przekłada się na wyższą dokładność oraz efektywność końcowego modelu klasyfikacji.

### Architektura sieci i użyte warstwy

**Architektura konwolucyjnej sieci neuronowej**

**Warstwa konwolucyjna**

Warstwa konwolucyjna jest jednym z kluczowych komponentów sieci neuronowych stosowanych w głębokim uczeniu, szczególnie w przetwarzaniu obrazów i rozpoznawaniu wzorców. Jej główna funkcja polega na wyodrębnieniu cech z przetwarzanych danych, takich jak obrazy, poprzez zastosowanie operacji zwanej konwolucją. W przeciwieństwie do zwykłych, gęsto połączonych warstw sieci neuronowych, warstwa konwolucyjna zachowuje przestrzenną hierarchię danych wejściowych, co jest szczególnie ważne w przypadku obrazów, gdzie lokalizacja i kontekst pikseli są istotne.

Działanie warstwy konwolucyjnej można opisać w następujących krokach:

1. Inicjalizacja filtrów (jąder konwolucyjnych): Warstwa konwolucyjna składa się z zestawu uczonych filtrów (również znanych jako jądra konwolucyjne), które mają niewielkie, ale ustalone wymiary (np. 3x3, 5x5 pikseli). Każdy filtr jest odpowiedzialny za wykrywanie określonych cech, takich jak krawędzie, kształty lub tekstury.
2. Przesuwanie filtrów przez obraz: Każdy filtr jest stosowany do obrazu (lub poprzedniej warstwy map cech) poprzez przesuwanie go przez całą przestrzeń wejściową. Proces ten jest znany jako operacja konwolucji.
3. Obliczanie wartości konwolucji: Dla każdej pozycji, na którą przesunięto filtr, obliczana jest suma iloczynów pikseli obrazu i wartości filtru. Jest to równoznaczne z obliczeniem iloczynu skalarnego pomiędzy pikselami obrazu a wartościami filtru.
4. Tworzenie map cech: Wynik konwolucji dla każdego filtra tworzy mapę cech, która jest esencją wykrytych wzorców w określonym regionie obrazu. Jeśli warstwa konwolucyjna zawiera wiele filtrów, każdy z nich generuje oddzielną mapę cech.
5. Zastosowanie przesunięcia (ang. stride) i dopełnienia (ang. padding): Przesunięcie określa, o ile pikseli filtr jest przesuwany za każdym razem. Dopełnienie polega na dodaniu dodatkowych pikseli (zwykle o wartości zero) do koła obrazu, co pozwala na regulację wymiarów wynikowej mapy cech.

Podsumowując, warstwa konwolucyjna w sieciach neuronowych jest fundamentalna dla analizy danych przestrzennych, takich jak obrazy. Dzięki swojej zdolności do wykrywania lokalnych cech i zachowania przestrzennych relacji między pikselami, stanowi kluczowy element w architekturach głębokiego uczenia stosowanych w rozpoznawaniu obrazów i wizji komputerowej.

**Max Pooling**

Warstwa max pooling, szeroko stosowana w sieciach konwolucyjnych (CNNs), odgrywa istotną rolę w procesie głębokiego uczenia, zwłaszcza w analizie wizualnej i przetwarzaniu obrazów. Jej głównym zadaniem jest redukcja wymiarów map cech wygenerowanych przez poprzednie warstwy konwolucyjne, co przyczynia się do zmniejszenia ilości parametrów i obliczeń w modelu. Proces ten nie tylko zwiększa efektywność obliczeniową sieci, ale także pomaga w zapobieganiu przeuczeniu, zachowując przy tym najbardziej istotne cechy w danych.

Działanie warstwy max pooling odbywa się w kilku krokach:

1. Definiowanie rozmiaru okna i przesunięcia (stride): Na początku ustalany jest rozmiar okna poolingowego, często będącego kwadratem (np. 2x2 lub 3x3 piksele), oraz przesunięcie, które określa, o ile pikseli okno jest przesuwane podczas każdej operacji.
2. Przesuwanie okna po mapie cech: Okno max pooling jest przesuwane przez mapę cech, uzyskaną z warstwy konwolucyjnej. Przesuwa się ono w kierunkach pionowym i poziomym zgodnie z zdefiniowanym przesunięciem.
3. Ekstrakcja maksymalnej wartości: W każdym oknie poolingowym wybierana jest największa wartość spośród wartości pikseli. Ta operacja pozwala na redukcję rozmiaru danych, zachowując jednocześnie kluczowe informacje.
4. Formowanie zredukowanej mapy cech: Rezultatem jest nowa mapa cech o zmniejszonych wymiarach, która składa się z wartości maksymalnych z każdego okna poolingowego. Ta zredukowana mapa cech nadal zawiera najbardziej wyraźne i charakterystyczne cechy z poprzedniej mapy.
5. Utrzymanie ważnych informacji przy zmniejszeniu wymiarów: Wybór największych wartości w każdym oknie poolingowym pozwala na zachowanie najważniejszych cech, takich jak krawędzie i tekstury, podczas jednoczesnego zmniejszenia rozmiaru danych. Jest to korzystne w kontekście rozpoznawania wzorców, gdzie dokładne położenie cech jest mniej istotne niż ich obecność.

Warstwa Max Pooling jest szczególnie efektywna w redukcji ilości parametrów i obliczeń w sieci, co prowadzi do szybszego uczenia i mniejszego ryzyka przeuczenia. Ponadto, dzięki redukcji wymiarowości, sieć staje się bardziej odporna na drobne przesunięcia i zniekształcenia w danych wejściowych, co jest kluczowe w wielu zastosowaniach wizyjnych.

**Leaky RelU**

Funkcja aktywacyjna Leaky ReLU (Rectified Linear Unit) jest wariacją standardowej funkcji ReLU, stosowaną w sieciach neuronowych, szczególnie przy głębokim uczeniu. Standardowa funkcja ReLU jest zdefiniowana jako , co oznacza, że dla wszystkich dodatnich wartości wejściowych zwraca wartość równą wejściu, a dla ujemnych wartości zwraca zero. Jednakże, ReLU ma pewną wadę znaną jako problem "umierających neuronów", gdy neuron zwraca zero dla wszystkich danych wejściowych, traci on swoją zdolność do dalszego uczenia.

Leaky ReLU ma na celu rozwiązanie tego problemu, wprowadzając małą stałą wartość dla ujemnych wartości wejściowych, zamiast zwracać zero. Jest zdefiniowana jako , gdzie *α* jest małą stałą (zazwyczaj bliską 0, np. 0.01). Dzięki temu, nawet gdy wartości wejściowe są ujemne, funkcja aktywacyjna nadal przekazuje informację, co pomaga w uniknięciu problemu umierających neuronów.

Wybór Leaky ReLU jest uzasadniony następującymi powodami:

* Zapobieganie Umieraniu Neuronów: Poprzez umożliwienie przepływu małej wartości gradientu dla ujemnych wartości wejściowych, Leaky ReLU zapobiega problemowi umierających neuronów, co jest ograniczeniem standardowej funkcji ReLU.
* Utrzymanie Gradientu: Leaky ReLU pomaga w utrzymaniu gradientu w sieci, co jest istotne dla procesu uczenia, szczególnie w głębokich sieciach.
* Poprawa Procesu Uczenia: Dzięki lepszemu przepływowi informacji i gradientów, Leaky ReLU może przyczynić się do szybszej i bardziej efektywnej konwergencji podczas treningu sieci.
* Zastosowanie w Różnych Zadaniach: Leaky ReLU jest uniwersalna i może być stosowana w różnych rodzajach sieci neuronowych, w tym w sieciach konwolucyjnych i rekurencyjnych.

Podsumowując, Leaky ReLU oferuje rozwiązanie dla pewnych ograniczeń ReLU, zachowując jednocześnie prostotę i efektywność, co sprawia, że jest wartościowym wyborem funkcji aktywacyjnej w wielu architekturach sieci neuronowych.

**Dropout**

Dropout to technika regularyzacji stosowana w sieciach neuronowych, mająca na celu zapobieganie nadmiernemu dopasowaniu (overfitting) modelu do danych treningowych. Overfitting jest typowym problemem w uczeniu maszynowym, który występuje, gdy model zbytnio "zapamiętuje" dane treningowe, tracąc zdolność do generalizowania na nowe, nieznane dane. Dropout efektywnie radzi sobie z tym wyzwaniem poprzez losowe "wyłączanie" (ignorowanie) niektórych neuronów w sieci podczas procesu treningu.

Dropout działa w następujący sposób:

1. Losowe deaktywowanie Neuronów: W każdej iteracji treningu, losowo wybierane neurony (wraz z ich połączeniami) są tymczasowo "wyłączane" z sieci. To znaczy, ich wartości aktywacji są ustawiane na zero. Procent neuronów, które są wyłączane, jest parametrem, który można dostosować (np. 50% neuronów może być wyłączanych w każdej iteracji).
2. Redukcja overfittingu: Przez losowe wyłączanie neuronów, sieć jest zmuszana do niepolegania na żadnym konkretnym zestawie cech. To zmusza model do uczenia się bardziej odpornej reprezentacji danych, która nie jest zależna od konkretnych cech danych treningowych.
3. Efekt Ensemble: Dropout wprowadza pewien rodzaj efektu ensemble w sieci neuronowej. Każdy zestaw wyłączonych neuronów może być traktowany jako osobny model, a końcowe wyniki są jakby średnią z różnych "wcieleń" sieci.

Inne techniki regularyzacji, takie jak L1 czy L2, polegają na dodawaniu kary do funkcji straty za duże wagi w sieci. Dropout natomiast działa na poziomie architektury sieci, losowo zmieniając aktywność neuronów, co pomaga w tworzeniu bardziej zrównoważonej i odpornej sieci. Głównym celem zastosowania dropoutu jest pomoc w stworzeniu modelu, który lepiej generalizuje na nowe dane, nie tracąc przy tym zdolności do nauki złożonych wzorców z danych treningowych. Jest to innymi słowy technika regularyzacji w uczeniu głębokim, zapobiegającą problemowi przeuczania dzięki losowemu wyłączaniu neuronów podczas treningu.

**Normalizacja wsadowa**

Jest technika stosowana w procesie uczenia sieci neuronowych, mającą na celu poprawę szybkości, wydajności i stabilności procesu uczenia. Wprowadzona w celu adresowania problemu "internal covariate shift" (wewnętrznej zmiany rozkładu danych), technika ta polega na normalizacji danych wejściowych dla każdej warstwy w sieci neuronowej. Normalizacja ta jest przeprowadzana nie tylko na początku całego procesu uczenia, ale także dynamicznie w trakcie treningu dla każdego wsadu (batch) danych.

Działanie normalizacji wsadowej prezentuje się w następujący sposób:

1. Normalizacja mini-wsadów: W normalizacji wsadowej, wejścia do warstwy sieci są normalizowane osobno dla każdego mini-wsadu. Oznacza to, że średnia i wariancja są obliczane i stosowane dla każdego mini-wsadu, a nie dla całego zestawu danych treningowych.
2. Stabilizacja średniej i wariancji: Normalizacja polega na dostosowaniu danych tak, aby miały średnią równą zero i wariancję równą jeden. Jest to osiągane poprzez odjęcie średniej mini-wsadu od każdego wejścia, a następnie podzielenie wyniku przez odchylenie standardowe mini-wsadu.
3. Wprowadzenie parametrów skalowania i przesunięcia: Po znormalizowaniu danych, normalizacja wsadowa wprowadza dwa parametry, które są uczone w procesie treningu: parametr skalowania i parametr przesunięcia. Pozwalają one sieci na przywrócenie potrzebnej nieliniowości, która mogłaby zostać utracona w wyniku normalizacji.

W przeciwieństwie do Dropout, który losowo wyłącza neurony, normalizacja wsadu skupia się na regulowaniu rozkładu danych wejściowych dla każdej warstwy.

Normalizacja wsadowa często jest stosowana razem z innymi technikami regularyzacji, jak Dropout, aby uzyskać jeszcze lepsze wyniki. Poprzez stabilizację rozkładów danych wejściowych, normalizacja pozwala na stosowanie wyższych szybkości uczenia, co może znacznie przyspieszyć proces treningu sieci. Sieci wykorzystujące normalizację wsadową często osiągają lepszą wydajność, ponieważ technika ta zmniejsza problem znikającego gradientu oraz pomaga w lepszej generalizacji. Normalizacja zmniejsza także zależność od doboru początkowych wartości wag i pozwala na bardziej elastyczne stosowanie szybkości uczenia, co ułatwia proces optymalizacji sieci neuronowej.

**Spłaszczanie**

Spłaszczanie to prosta, ale kluczowa operacja stosowana w procesie budowy sieci neuronowych, zwłaszcza tych używanych do przetwarzania obrazów, jak sieci konwolucyjne (CNN). Proces ten polega na przekształceniu wielowymiarowych danych wejściowych (np. obrazów) w jednowymiarowy wektor. Ta operacja jest szczególnie ważna w przejściu od warstw konwolucyjnych i poolingowych do gęstych warstw sieci (fully connected layers), które wymagają danych wejściowych w formie jednowymiarowego wektora.

Działanie spłaszczenia:

1. Funkcja bierze wielowymiarową mapę cech (np. wynikającą z kolejnych warstw konwolucyjnych i poolingowych w CNN) i 'spłaszcza' ją, tworząc jednowymiarowy wektor. Na przykład, jeśli mapa cech ma wymiary 3x3x32, zostanie przekształcona w wektor o długości 288 (3 \* 3 \* 32).
2. Spłaszczone dane są następnie przekazywane do jednej lub więcej gęstych warstw (fully connected layers), które wymagają danych w formie jednowymiarowej. W tych warstwach każdy neuron jest połączony z każdym elementem wejściowego wektora danych.

Funkcja ta różni się od innych operacji przetwarzania danych, ponieważ nie wprowadza żadnych zmian w wartościach danych; jedynie zmienia ich strukturę. Nie wpływa to na informacje zawarte w danych, ale umożliwia ich dalsze przetwarzanie w sieci neuronowej. W CNN, warstwy konwolucyjne i poolingowe efektywnie wykrywają lokalne cechy obrazu. Spłaszczenie umożliwia przekształcenie tych cech w formę, którą można przetworzyć za pomocą standardowych, gęstych warstw sieciowych.

**Sigmoid**

Funkcja sigmoid, znana również jako funkcja logistyczna, jest jedną z najbardziej rozpowszechnionych funkcji aktywacyjnych używanych w sieciach neuronowych, szczególnie w zadaniach związanych z klasyfikacją binarną. Jest to funkcja S-kształtna, która skutecznie przekształca wartości wejściowe, mogące przyjmować dowolne liczby rzeczywiste, na wartości wyjściowe w zakresie od 0 do 1. Równanie tej funkcji to **,** gdzie e jest podstawą logarytmu naturalnego, a x jest wartością wejściową. Dzięki tej charakterystyce, wyjście funkcji sigmoid może być interpretowane jako prawdopodobieństwo. W kontekście klasyfikacji binarnej, gdzie model musi zdecydować, do której z dwóch klas należy dany przykład, wyjście sigmoidalne wskazuje prawdopodobieństwo przynależności do jednej z tych klas.

W przeciwieństwie do innych popularnych funkcji aktywacyjnych, takich jak ReLU czy softmax, sigmoid jest szczególnie użyteczny w sytuacjach, gdzie wynik modelu musi być ograniczony do przedziału [0, 1], co jest idealne dla modelowania prawdopodobieństw. W przypadku ReLU, która jest często stosowana w ukrytych warstwach sieci neuronowych, głównym celem jest wprowadzenie nieliniowości do modelu, co jest korzystne w wielu problemach uczenia maszynowego, ale nie jest odpowiednie do bezpośredniego modelowania prawdopodobieństw. Z kolei softmax, choć podobna w swoim działaniu do sigmoidu, jest bardziej odpowiednia w scenariuszach klasyfikacji wieloklasowej, gdzie model musi przypisać prawdopodobieństwo do wielu klas jednocześnie.

Sigmoid jest więc standardowym wyborem dla ostatniej warstwy w sieciach neuronowych zajmujących się klasyfikacją binarną. Jego zdolność do ograniczenia wyników do zakresu [0, 1] oraz do modelowania ich jako prawdopodobieństw sprawia, że jest niezastąpiony w wielu aplikacjach, gdzie wymagane jest probabilistyczne podejście. Dzięki sigmoidowi, wyniki modelu są łatwe do interpretacji i mogą służyć do podejmowania informowanych decyzji w oparciu o poziom pewności modelu co do swoich przewidywań.

### Dobór hiperparametrów

W procesie konstruowania i optymalizacji sieci neuronowych z wykorzystaniem różnych warstw i funkcji, jak wspomniane wcześniej konwolucje, pooling, dropout, normalizacje wsadów, oraz funkcje aktywacyjne takie jak ReLU, Leaky ReLU, sigmoid i softmax, istnieje wiele hiperparametrów, które mogą być dostosowywane w celu poprawy wydajności modelu. Oto niektóre z kluczowych hiperparametrów, którymi można manipulować:

1. Liczba i rozmiar warstw: Decyzje dotyczące liczby warstw konwolucyjnych, poolingowych, gęstych (fully connected) oraz ich rozmiaru mają znaczący wpływ na zdolność modelu do nauki. Zwiększanie liczby warstw i neuronów może zwiększyć zdolność sieci do modelowania złożonych wzorców, ale również zwiększa ryzyko przeuczenia (overfitting).
2. Rozmiar filtrów i stride w warstwach konwolucyjnych: Rozmiar filtrów (np. 3x3, 5x5) oraz krok przesuwania filtra (stride) wpływają na to, jak sieć postrzega dane wejściowe. Mniejsze filtry mogą wyłapywać drobniejsze detale, ale większe filtry mogą być lepsze do uchwycenia większych wzorców.
3. Liczba filtrów w warstwach konwolucyjnych: Większa liczba filtrów zwiększa głębokość map cech, co może pozwolić na wykrycie większej liczby cech w danych, ale także zwiększa złożoność obliczeniową modelu.
4. Parametry poolingu: W przypadku warstw poolingowych, można dostosować rozmiar okna poolingowego oraz krok przesunięcia. Decyzje te wpływają na stopień redukcji przestrzennych wymiarów danych.
5. Współczynnik dropoutu: W warstwach dropout należy określić współczynnik dropoutu, czyli procent neuronów wyłączanych podczas treningu. Zbyt wysoki współczynnik może prowadzić do niedouczenia (underfitting), podczas gdy zbyt niski może nie zapewniać wystarczającej regularyzacji.
6. Szybkość uczenia: Jest to jeden z najważniejszych hiperparametrów w procesie uczenia się sieci neuronowej. Zbyt wysoka szybkość uczenia może powodować, że model będzie przeskakiwał optymalne rozwiązania, podczas gdy zbyt niska szybkość uczenia może znacząco wydłużyć czas potrzebny do nauki lub doprowadzić do zatrzymania się w minimach lokalnych.
7. Algorytm optymalizacji: Wybór algorytmu optymalizacji (np. SGD, Adam, RMSprop) ma znaczący wpływ na proces treningu sieci. Różne algorytmy mają różne sposoby aktualizacji wag sieci w procesie uczenia.
8. Rozmiar wsadu: Określa liczbę próbek danych przetwarzanych w sieci jednocześnie. Większe batche mogą przyspieszyć trening, ale wymagają więcej pamięci i mogą wpłynąć na jakość modelu.
9. Inicializacja wag: Wybór metody inicjalizacji wag (np. Xavier, He initialization) może wpłynąć na początkową fazę treningu sieci, zwłaszcza w głębokich architekturach.
10. Liczba epok treningowych: Określa, ile razy cały zestaw danych treningowych zostanie użyty do aktualizacji wag sieci. Więcej epok może prowadzić do lepszego nauczenia, ale również do ryzyka przeuczenia.

Każdy z tych hiperparametrów ma wpływ na sposób, w jaki sieć uczy się z danych i jak dobrze może generalizować na nowych danych. Dostosowywanie tych parametrów często wymaga eksperymentowania i strojenia, aby znaleźć optymalne ustawienia dla konkretnego problemu.

## Metryki i sposób walidacji algorytmu

Poniżej wymieniono metryki pozwalające na wszechstronną ocenę wydajności modelu, uwzględniając różne aspekty klasyfikacji, takie jak zdolność do prawidłowego identyfikowania poszczególnych klas, błędy w przewidywaniach oraz ogólną dokładność. W kontekście medycznym, gdzie błędna diagnoza może mieć poważne konsekwencje, zastosowanie tych metryk jest szczególnie istotne.

1. Straty (Loss)

Określa błąd między przewidywaniami modelu a rzeczywistymi etykietami.

W sieciach neuronowych często stosuje się funkcję straty, jak entropia krzyżowa (cross-entropy).

1. Macierz pomyłek (Confusion Matrix)

Tablica, która ilustruje liczbę fałszywych i prawdziwych wyników dla każdej klasy, rozróżniając między pozytywnymi i negatywnymi predykcjami.

1. ROC (Receiver Operating Characteristic)

Opisuje zależność między prawdziwie pozytywnymi a fałszywie pozytywnymi wynikami klasyfikacji.

Często przedstawiane jako wykres.

1. AUC (Area Under the Curve)

Opisuje obszar pod krzywą ROC.

Im większa wartość AUC (najlepiej blisko 1), tym lepsza zdolność modelu do rozróżniania między klasami.

1. Dokładność (Accuracy)

Opisuje ogólną zdolność modelu do poprawnego klasyfikowania przypadków.

1. Czułość (Sensitivity)

Mierzy zdolność modelu do prawidłowego identyfikowania rzeczywistych pozytywów.

1. Swoistość (Specificity)

Mierzy zdolność modelu do prawidłowego identyfikowania rzeczywistych negatywów.

1. Precyzja (Precision)

Określa, jaki odsetek przewidzianych pozytywnych przypadków jest rzeczywiście pozytywny.

1. F1-Score

Jest średnią harmoniczną precyzji i czułości, przydaje się w sytuacjach, gdy klasa jest niezbalansowana.

# Implementacja algorytmu

Podział danych na podzbiory (Cross walidacja)

Segmentacja

Augmentacja

Przygotowanie danych

Architektura sieci neuronowej (conv, leaky relu, max / min pooling, dropout)

Hiperparametry

Trenowanie i testowanie

Walidacja

# Prezentacja algorytmu

Wyniki

# Podsumowanie

# Bibliografia

https://www.aurorabiomed.com/pros-cons-covid-19-antigen-rt-pcr-antibody-test/

https://www.nature.com/articles/s43856-022-00147-y

<https://www.nature.com/articles/s41598-023-48892-x>

1. Shastri, S., Kansal, I., Kumar, S. et al. CheXImageNet: a novel architecture for accurate classification of Covid-19 with chest x-ray digital images using deep convolutional neural networks. Health Technol. 12, 193–204 (2022). https://doi.org/10.1007/s12553-021-00630-x

1. Kumar S, Shastri S, Mahajan S, et al. LiteCovidNet: A lightweight deep neural network model for detection of COVID-19 using X-ray images. Int J Imaging Syst Technol. 2022;1‐17. DOI: <https://doi.org/10.1002/ima.22770>

@misc{tabik2020covidgr,

title={COVIDGR dataset and COVID-SDNet methodology for predicting COVID-19 based on Chest X-Ray images},

author={S. Tabik and A. Gómez-Ríos and J. L. Martín-Rodríguez and I. Sevillano-García and M. Rey-Area and D. Charte and E. Guirado and J. L. Suárez and J. Luengo and M. A. Valero-González and P. García-Villanova and E. Olmedo-Sánchez and F. Herrera},

year={2020},

eprint={2006.01409},

archivePrefix={arXiv},

primaryClass={eess.IV}

}

<https://ieeexplore.ieee.org/document/9254002>

<https://github.com/jeffheaton/app_deep_learning/blob/main/t81_558_class_05_2_cnn.ipynb>

<https://github.com/jeffheaton/app_deep_learning/blob/main/t81_558_class_03_2_pytorch.ipynb>

<https://github.com/seungjunlee96/U-Net_Lung-Segmentation>

<https://www.mdpi.com/2071-1050/15/7/5930>

<https://medium.com/@kyan7472/these-are-the-5-best-pre-trained-neural-networks-23798e61a043>

<https://link.springer.com/article/10.1007/s42979-021-00695-5>

https://bmcpulmmed.biomedcentral.com/articles/10.1186/s12890-022-02068-x#Tab4