

Łukasz POLOCZEK<sup>1</sup>, Michał WILKOSZ<sup>2</sup>, Piotr CZECH<sup>3</sup>,  
Mariola SATERNUS<sup>4</sup>, Henryk KANIA<sup>5</sup>

Opiekunowie naukowci: Piotr CZECH<sup>6</sup>, Mariola SATERNUS<sup>7</sup>,  
Henryk KANIA<sup>8</sup>

DOI: <https://doi.org/10.53052/9788366249837.20>

## WYKORZYSTANIE SZTUCZNYCH SIECI NEURONOWYCH TYPU MLP DO PREDYKCJI ZANIECZYSZCZENIA POWIETRZA NA PODSTAWIE DANYCH POGODOWYCH ZE STACJI POMIAROWEJ

**Streszczenie:** Artykuł miał na celu przedstawienie wyników badań dotyczących predykcji wartości zanieczyszczenia powietrza na podstawie zarejestrowanych na stacji pomiarowej danych pogodowych. W badaniach wykorzystano sztuczne sieci neuronowe typu perceptron wielowarstwowy. Przeanalizowano możliwość wykorzystania do obliczeń różnych danych wejściowych, w tym dwutlenku siarki (SO<sub>2</sub>), dwutlenku azotu (NO<sub>2</sub>), tlenku azotu (NO<sub>x</sub>), tlenku węgla (CO), ozonu (O<sub>3</sub>), benzenu (C<sub>6</sub>H<sub>6</sub>), pyłu zawieszonego PM10, pyłu zawieszonego PM2.5, ciśnienia atmosferycznego, wilgotności względnej, temperatury, kierunku wiatru, prędkości wiatru, promieniowania RAD, opadu atmosferycznego i promieniowania UVB. Badania zrealizowano w ramach projektu Politechnika Śląska jako Centrum Nowoczesnego Kształcenia opartego o badania i innowacje POWR-03.05.00-00-Z098/17-00.

**Słowa kluczowe:** sieć neuronowa, perceptron wielowarstwowy, zanieczyszczenie powietrza

---

<sup>1</sup> Politechnika Śląska, Wydział Automatyki, Elektroniki i Informatyki, specjalność: macrofaculty – data science

<sup>2</sup> Politechnika Śląska, Wydział Automatyki, Elektroniki i Informatyki, specjalność: macrofaculty – data science

<sup>3</sup> Politechnika Śląska, Wydział Transportu i Inżynierii Lotniczej, email: piotr.czech@polsl.pl

<sup>4</sup> Politechnika Śląska, Wydział Inżynierii Materiałowej, email: mariola.saternus@polsl.pl

<sup>5</sup> Politechnika Śląska, Wydział Inżynierii Materiałowej, email: henryk.kania@polsl.pl

<sup>6</sup> Prof. dr hab. inż., Politechnika Śląska, Wydział Transportu i Inżynierii Lotniczej, email: piotr.czech@polsl.pl

<sup>7</sup> Dr hab. inż., prof. PŚ, Politechnika Śląska, Wydział Inżynierii Materiałowej, email: mariola.saternus@polsl.pl

<sup>8</sup> Dr hab. inż., Politechnika Śląska, Wydział Inżynierii Materiałowej, email: henryk.kania@polsl.pl

## APPLICATION OF MLP NEURAL NETWORKS FOR PREDICTION OF AIR POLLUTION BASED ON WEATHER DATA FROM A MEASURING STATION

**Summary:** The aim of the article was to present the results of research on air pollution value prediction based on the weather data recorded at the measuring station. Multi-layer perceptron were used in the research. The possibility of using various input data for calculations was analysed, including sulfur dioxide (SO<sub>2</sub>), nitrogen dioxide (NO<sub>2</sub>), nitrogen oxide (NO<sub>x</sub>), carbon monoxide (CO), ozone (O<sub>3</sub>), benzene (C<sub>6</sub>H<sub>6</sub>), PM10 particulate matter, PM2.5 particulate matter, atmospheric pressure, humidity relative temperature, wind direction, wind speed, RAD radiation, precipitation and UVB radiation. The research was carried out as part of the project of the Silesian University of Technology as a Modern Education Center based on research and innovation POWR-03.05.00-00-Z098 / 17-00.

**Keywords:** neural network, multilayer perceptron, air pollution

### 1. Wprowadzenie dotyczące sztucznych sieci neuronowych

Eksperyment badawczy polegał na stworzeniu modelu perceptronu wielowarstwowego, który będzie potrafił na podstawie danych wejściowych ze stacji pomiarowej oszacować wartości zanieczyszczeń. Na początku należy przybliżyć pojęcia związane z sieciami neuronowymi, które były niezbędne do zrozumienia podjętej tematyki badań.

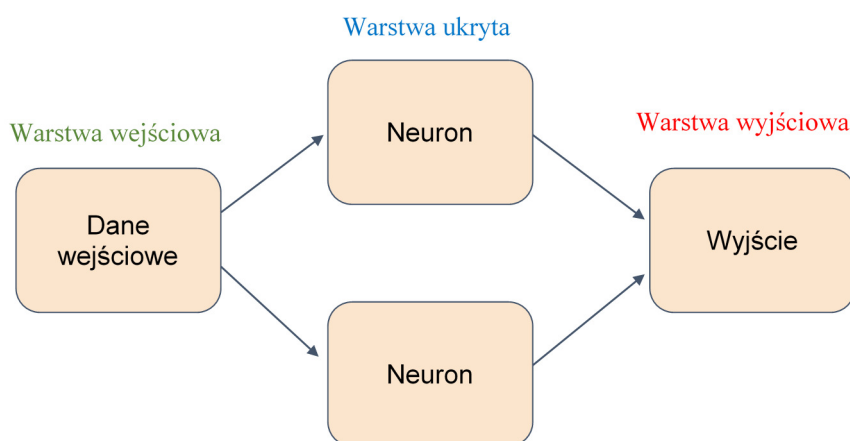
Badania nad sztucznymi sieciami neuronowymi rozpoczęły się od próby formalnego opisu procesów zachodzących w komórkach nerwowych mózgu. Następnie starano się odwzorować wybrane własności, tworząc pierwsze sztuczne sieci na podstawie biologicznej sieci neuronowej. Ludzki mózg składa się z neuronów przetwarzających informacje, które są następnie zapamiętywane w połączeniach synaptycznych. Przetwarzanie informacji odbywa się w sposób równoległy, co daje przewagę przy detekcji mowy lub rozpoznawaniu obrazów nad szeregowo połączonymi procesorami komputera. Model sztucznego neuronu wykorzystywanego do stworzenia sztucznej sieci neuronowej został oparty na budowie komórki nerwowej ludzkiego układu nerwowego. Neuron charakteryzuje się otaczającymi ciało wypustkami zwanymi dendrytami, które tworzą wejścia komórki. W ciele komórki informacja w postaci impulsów elektrycznych jest sumowana, a następnie generowany jest sygnał wyjściowy. Aksony to włókna wychodzące z ciała komórki, które rozgałęziają się w tzw. drzewka aksonowe. Koniec aksonu łączy się z dendrytami następnej komórki za pomocą połączenia zwanego synapsą. To od siły połączenia synaptycznego zależy czy neuron będzie aktywny, czy pozostanie w spoczynku. Dzięki sieci komórek mózgu może realizować złożone funkcje takie jak zapamiętywanie czy twórcze myślenie.

Sztuczne sieci neuronowe to ogólna nazwa struktur matematycznych i ich programowych lub sprzętowych modeli, realizujących obliczenia lub przetwarzanie sygnałów poprzez rzędy elementów przetwarzających (neuronów), wykonujących pewną podstawową operację na swoim wejściu. Sztuczne sieci neuronowe są związane z pojęciem sztucznej inteligencji i mogą być wykorzystywane do takich zadań jak: prognozy giełdowe, prognozowanie sprzedaży, poszukiwanie ropy

naftowej, interpretacja badań biologicznych, prognoza cen, analiza badań medycznych, planowanie remontów maszyn, planowanie postępów w nauce, analiza problemów produkcyjnych i inne.

Perceptron wielowarstwowy (z ang. Multilayer Perceptron, w skrócie MLP), to popularny typ sztucznej sieci neuronowej. Składa się ona zwykle z jednej warstwy wejściowej, kilku warstw ukrytych oraz jednej warstwy wyjściowej neuronów. Często jest nazywany siecią w pełni połączoną, z uwagi na istnienie architektury połączeń pomiędzy neuronami, w której wszystkie wyjścia warstwy wcześniejszej połączone są z odpowiednimi wejściami każdego neuronu warstwy następnej. Zasada działania neuronu polega na przemnożeniu wartości wejściowych i wag poszczególnych połączeń, a następnie zsumowaniu otrzymanych wyników. W dalszej kolejności korzystając z wybranej funkcji aktywacyjnej, takiej jak na przykład funkcja sigmoidalna lub tangens hiperboliczny, neuron tworzy wartość wyjściową. Jest ona równocześnie wartością wejściową neuronów kolejnej warstwy, jeżeli taka istnieje. Ustalenie właściwej liczby warstw ukrytych oraz liczby neuronów znajdujących się w poszczególnych warstwach jest trudnym zagadnieniem, które musi rozwiązać twórca sieci neuronowej. Po określeniu liczby warstw oraz liczby neuronów w każdej warstwie należy dobrać wartości wag oraz bias, który można określić, jako stałą wartość dodawaną do wartości wejściowej danego neuronu w celu lepszego dopasowania danych do funkcji aktywacyjnej. Parametry te wybiera się w sposób zapewniający poprawę działania sieci, przy czym początkowa wartość wag jest zazwyczaj losowa, a bias jest równy 0. Podstawą do modyfikacji parametrów zapewniającej wydajniejszy model są zebrane przykłady, czyli dane wejściowe wraz z podanymi dla nich prawidłowymi rozwiązaniami. Taka technika uczenia maszynowego jest określana, jako uczenie nadzorowane, ponieważ oprócz danych wejściowych znane są oczekiwane dane wyjściowe i na ich podstawie tworzy się model perceptronu wielowarstwowego.

Na rysunku 1 przedstawiono schematycznie wygląd sztucznej sieci neuronowej typu MPL.



Rysunek 1. Sieć neuronowa typu MLP

Proces uczenia sztucznych sieci neuronowych można zdefiniować, jako dopasowywanie parametrów modelu reprezentowanego przez sieć do dostępnych

danych uczących. Żeby określić wydajność sieci neuronowej o określonej konfiguracji wag należy ją uruchomić dla wszystkich przypadków uczących i porównać wartości otrzymywane na wyjściu sieci z wartościami zadanymi jako wzorce poprawnych rozwiązań. Wartość określającą jakość modelu wyznacza się za pomocą funkcji zwanej funkcją straty lub funkcją kosztu (z ang. loss function). Służy ona do obliczenia różnicy między prognozowanym wynikiem a wynikiem prawdziwym. Jest ona miarą błędu popełnionego przez sieć. Im mniejsza wartość tej funkcji, tym model jest lepszy. Zmiany w sieci neuronowej polegają na dopasowaniu wag i wartości biasu tak, aby wartość funkcji strat miała jak najmniejszą wartość. Często stosowaną funkcją kosztu jest suma kwadratów błędów. Przy jej stosowaniu wyznaczone różnice wartości wyliczonych przez sieć neuronową i wartości poprawnych, zawartych jako wzorzec, są podnoszone do kwadratu, a następnie sumowane. Najpopularniejszą techniką wykorzystywaną do uczenia sieci neuronowej jest metoda wstecznej propagacji błędu (z ang. backpropagation) z różnymi metodami optymalizacyjnymi.

Propagacja wstecz jest pragmatyczną próbą rozłożenia odpowiedzialności za błąd wyniku na poszczególne wartości wejściowe za pomocą wag. W rzeczywistości jest to prosta implementacja reguły łańcucha z pochodnych, która daje możliwość obliczenia wszystkich wymaganych pochodnych cząstkowych funkcji złożonej. Obliczany jest gradient, czyli wektor w przestrzeni wag funkcji straty sieci neuronowej. Wektor ten określa linię przechodzącą przez bieżący punkt i wyznaczającą kierunek, w którym spadek wartości błędu jest najszybszy. Gradient wskazuje kierunek najszybszego wzrostu funkcji, a w przypadku zmiany znaków składowych na przeciwne, czyli pomnożeniu przez  $-1$ , kierunek jej najszybszego spadku nazywanego antygradientem. Metoda wstecznej propagacji błędu opisuje zmianę wag dowolnych połączeń neuronów w sąsiednich warstwach sieci. Bazuje na wykorzystaniu sumy kwadratów błędów (lub innej funkcji straty) i jest wykorzystywana w metodach optymalizacji modelu uczenia. Najbardziej znaną grupą algorytmów optymalizacyjnych czerpiących korzyści z techniki wstecznej propagacji błędu są metody gradientowe takie jak gradient prosty lub stochastyczny spadek gradientu.

Uogólniając wcześniejsze rozważania, uczenie sieci neuronowych typu MLP jest to optymalizacja wartości wag w celu minimalizacji błędu popełnianego przez sieć.

## 2. Analiza danych pomiarowych

Celem przeprowadzonych badań była próba predykcji zanieczyszczenia powietrza na podstawie danych pogodowych ze stacji pomiarowej. Jako narzędzie pomocne do tego celu wybrano jedną z metod sztucznej inteligencji w postaci sztucznej sieci neuronowej typu MLP.

Pierwszym etapem przeprowadzonych badań była analiza danych potrzebnych do stworzenia modelu sztucznej sieci neuronowej. W przeprowadzonych eksperymentach wykorzystano dane ze stacji pomiarowych z dolnośląskiego systemu monitoringu jakości powietrza. Do dyspozycji były pomiary z przedziału czasowego od 2005 do 2017 roku. Niestety w wielu plikach brakowało znaczącej liczby danych, dlatego skupiono się na tylko jednym zestawie danych pochodzących z jednej stacji monitoringu powietrza. Wybrano stację znajdującą się we Wrocławiu przy ulicy

Wybrzeża J. Conrada-Korzeniowskiego. Wyboru dokonano na podstawie porównania z innymi stacjami pomiarowymi, przyjmując za kryterium najmniejszą liczbę brakujących danych i dużą liczbę mierzonych parametrów. Dostępnymi parametrami dla tej stacji były dwutlenek siarki (SO<sub>2</sub>), dwutlenek azotu (NO<sub>2</sub>), tlenki azotu (NO<sub>x</sub>), tlenek węgla (CO), ozon (O<sub>3</sub>), benzen (C<sub>6</sub>H<sub>6</sub>), pył zawieszony PM10, pył zawieszony PM2.5, ciśnienie atmosferyczne, wilgotność względna, temperatura, kierunek wiatru, prędkość wiatru, promieniowanie RAD, opad atmosferyczny i promieniowanie UVB. We wszystkich analizowanych zbiorach danych największe braki, które uniemożliwiały korzystanie z danych znajdowały się w początkowych latach. Dla wybranej stacji najmniej braków było w ostatnich trzech latach pomiarów, czyli w przedziale 2015-2017. Okres ten poddano dalszym analizom. Równocześnie podjęto decyzje o uzupełnieniu pojedynczych braków przy wykorzystaniu dostępnych metod matematycznych.

Przykładowe dane zestawiono w tabeli 1.

Tabela 1. Przykładowe dane wejściowe do przeprowadzonych badań

Rok	2015	2015	2015	2015	2015
Miesiąc	1	1	1	1	1
Dzień	1	1	1	1	1
Godzina	0	1	2	3	4
Ciśnienie atmosferyczne	1012	1011	1011	1012	1012
Temperatura powietrza	-6,6	-7,4	-7,1	-6,5	-5,6
Promieniowanie RAD	0	0	0	0	0
Promieniowanie UVB	0	0	0	0	0
Prędkość wiatru	1,6	1,6	1,6	1,5	1,7
Wilgotność względna powietrza	62,8	69,6	72,3	71	69,2
Opady atmosferyczne	1,6	1,5	1,4	1,5	1,7
Kierunek wiatru	278	279	286	292	299
CO	1	1	1	1	1
NO <sub>2</sub>	18	20	20	21	22
NO <sub>x</sub>	20	25	25	24	25
O <sub>3</sub>	7	7	6	6	6
PM2.5	19	13	11	9	10
PM10	32	50	34	29	21
SO <sub>2</sub>	36	244	103	78	70
C <sub>6</sub> H <sub>6</sub>	5,01	5,25	4,74	4,67	4,77

Na podstawie przyjętego celu badań określono typ zadania, jakie miał projektowany model neuronowy. Zadaniem tym była regresja. Regresja jest to grupa algorytmów wykorzystywanych w uczeniu maszynowym. Jest to metoda statystyczna służąca określaniu związku pomiędzy różnymi wielkościami i przewidywaniu nieznanymi wartościami jednych wielkości na podstawie znanych wartości innych. Przykładem wykorzystania jest przewidywanie długości życia człowieka przy pomocy parametrów życiowych, przedstawionych w postaci wartości liczbowych. Rozwiązanie, które jest liczbą rzeczywistą i można je przedstawić w postaci funkcji ciągłej zawiera się w problematyce algorytmów regresji. W przeprowadzonych eksperymentach wykorzystano regresję wieloczynnikową, czyli badanie zależności pomiędzy zmienną zależną (zanieczyszczeniem powietrza) a większą liczbą zmiennych niezależnych. Zmienne niezależne to takie, które były wykorzystane od

początku procesu uczenia i za ich pomocą przewidywano wartości zmiennych zależnych.

Do stworzenia modelu sztucznej sieci neuronowej zastosowany został język programowania Python wraz z biblioteką Tensorflow, która jest wykorzystywana w uczeniu maszynowym i głębokich sieciach neuronowych. Z uwagi na złożoność obliczeń i duże wymagania zasobów komputerowych, w eksperymentach używano Google Colaboratory, czyli rozwiązania chmurowego pozwalającego uruchamiać kod programu w przeglądarce internetowej.

Jako dane do przewidywania wybrano zanieczyszczenie powietrza atmosferycznego tlenkiem azotu ( $\text{NO}_x$ ) oraz dwutlenkiem siarki ( $\text{SO}_2$ ). Wstępna analiza tych danych pozwoliła zauważyć, że dla tych zmiennych występuje wiele wartości znacznie odbiegających od większości pomiarów.

Zestawienie informacji dotyczących wybranych zmiennych zależnych pokazano w tabeli 2.

Tabela 2. Zestawienie informacji dotyczących wybranych danych pomiarowych

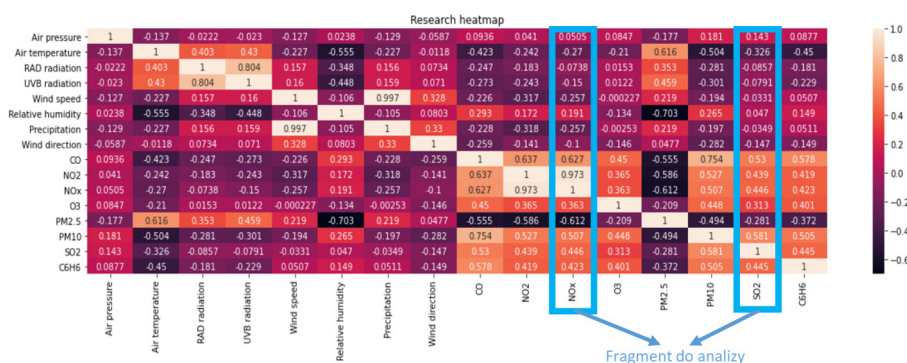
	$\text{NO}_x$	$\text{SO}_2$
Średnia arytmetyczna	36,404095	34,115177
Odchylenie standardowe	42,411309	25,118772
Wartość 75% całego zbioru	43,0	40,074700
Wartość maksymalna	735,0	383,221

Takie wyniki mogą świadczyć o błędach w pomiarze lub błędnym uzupełnieniu brakujących danych i konieczne jest ich usunięcie. W tym celu usunięto ze zbioru danych wiersze, dla których wartości  $\text{NO}_x$  i  $\text{SO}_2$  były większe niż  $100 \text{ ug/m}^3$ . Po usunięciu problematycznych danych, z 26304 początkowych wierszy pozostało 21657. Należy przy tym stwierdzić, że była to wystarczająca liczba potrzebna do stworzenia modelu regresji.

Dalsza analiza danych polegała na sprawdzeniu relacji pomiędzy wszystkimi zmiennymi zależnymi, czyli  $\text{NO}_x$  i  $\text{SO}_2$ , a zmiennymi niezależnymi, czyli parametrami, które posłużą do przewidywania wartości zanieczyszczeń powietrza atmosferycznego. Wykorzystując korelację metodą Spearmana, przeanalizowano siłę współzależności pomiędzy zmiennymi na podstawie mapy wartości odpowiednich współczynników. Wartości były z zakresu między -1 a 1. Wartość dodatnia opisuje to relacja, w której wzrost wartości jednej zmiennej definiuje wzrost wartości drugiej zmiennej. Analogicznie wartości ujemne bliższe -1 opisują wzrost wartości jednej zmiennej definiowany spadkiem drugiej. Wartości bliskie zeru wskazują na brak zależności.

Na rysunku 2 pokazano mapę korelacji Spearmana.

Po przeanalizowaniu mapy można stwierdzić, że dla obu wybranych zmiennych, których wartości miały być przewidywane, największą negatywną korelację wykazywał pył zawieszony  $\text{PM}_{2.5}$ . Warto zauważyć, że  $\text{NO}_2$  należy do grupy tlenków azotów  $\text{NO}_x$ , więc jego wartość korelacji jest prawie idealna (wartość równa 0,973). Dlatego też zdecydowano, że przy tworzeniu sieci neuronowych zarówno dla  $\text{NO}_x$  jak i  $\text{SO}_2$  ta zmienna nie będzie wykorzystywana, ponieważ może się to odbić na interpretowalności modelu.



Rysunek 2. Mapa korelacji Spearmana

Dodatkowo, zmienne w postaci promieniowanie RAD i promieniowania UVB nie były używane przy procesie modelowania sieci neuronowych. Jest to spowodowane tym, że duża część wartości tych zmiennych została uzupełniona sztucznie i mogło to wpłynąć niekorzystnie na prawdziwość działania modelu.

Do uczenia modelu  $\text{NO}_x$  nie były też wykorzystywane dane  $\text{SO}_2$  i analogicznie dla modeli neuronowych  $\text{SO}_2$  nie były używane  $\text{NO}_x$ . Wynika to z faktu, że zmienne te miały być przewidywane, a ich dodanie sztucznie zwiększyłoby już dużą liczbę zmiennych niezależnych wykorzystywanych do uczenia sieci.

### 3. Podział zbioru danych i przeskalowanie zmiennych

Kolejnym etapem badań był podział wybranych danych na zbiór treningowy, walidacyjny i testowy. Są one niezbędne do przeprowadzenia procesu uczenia oraz poprawnej oceny i analizy działania modelu neuronowego.

Zbiór treningowy to zbiór danych szkoleniowych. Jest to zbiór przykładów używanych podczas procesu uczenia i służy do dopasowania parametrów modelu, czyli wag połączeń neuronów. Uczenie na zbiorze testowym polega na dopasowaniu wag tak, aby wartości wyjściowe modelu zgadzały się z etykietowanymi wartościami zbioru treningowego.

Zbiór walidacyjny jest to zbiór przykładów używanych do dostrojenia hiperparametrów, na przykład takich jak liczba neuronów lub liczba powłok sieci. Zazwyczaj jest on częścią zbioru treningowego, ponieważ wraz z nim jest wykorzystywany podczas procesu uczenia.

Zbiór testowy stanowi zbiór niezależny od zbioru treningowego, służący jedynie do oceny wydajności sieci neuronowej. Ostateczny model przewiduje wartości przykładów ze zbioru testowego, a następnie są one porównywane z prawdziwymi wartościami przykładów, aby ocenić dokładność modelu.

W przeprowadzonych badaniach podział procentowy dla zadania regresji wyglądał następująco:

- zbiór treningowy: 70% wszystkich danych,
- zbiór walidacyjny: 20% danych zbioru treningowego,

- zbiór testowy: 30% wszystkich danych, czyli te dane, które nie zostały użyte w zbiorze treningowym.

Następnym etapem była normalizacja danych, czyli procedura wstępnej obróbki danych w celu umożliwienia ich wzajemnego porównywania i dalszej analizy. Polega ona na przekształceniu zmiennych tak, aby każda miała tę samą skalę, czyli ustalony punkt odniesienia. Taka technika ułatwia sieciom neuronowym proces uczenia. Jest wykorzystywana, gdy niekompatybilność pomiarów pomiędzy zmiennymi może mieć wpływ na osiągnięte wyniki. Najpopularniejszą metodą normalizacji jest skalowanie typu min-max. Polega ona na przeskalowaniu wartości zmiennych zazwyczaj do przedziału [0,1] wykorzystując wartość najmniejszą i największą podanej zmiennej według wzoru:

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)} \quad (1)$$

Inną metodą skalowania zmiennych jest standaryzacja. Zakłada ona, że wszystkie zmienne mają rozkład Gaussa. Skalowanie polega na odjęciu od oryginalnych danych pewnej wielkości, najczęściej średniej wartości zmiennej ( $\mu$ ) i podzieleniu przez odchylenie standardowe ( $\sigma$ ). W wyniku standaryzacji przekształcone wartości mają rozkład o średniej równej 0 i odchyleniu standardowym równym 1. To przekształcenie czyni wyniki analizy niezależne od jednostki pomiaru poszczególnych zmiennych, co ułatwia proces uczenia dla danych o rozkładzie Gaussa. Popularną nazwą tej metody jest tzw. Z-score.

$$x' = \frac{x - \mu}{\sigma} \quad (2)$$

#### 4. Przebieg tworzenia modelu perceptronu wielowarstwowego

Następnym etapem badań po przeskalowaniu wszystkich zmiennych wykorzystywanych do przewidywania zanieczyszczeń powietrza atmosferycznego było stworzenie modelu sztucznej sieci neuronowej. Do zbudowania architektury należy ustalić liczbę warstw sieci, liczbę neuronów znajdujących się w każdej warstwie, funkcję aktywacyjną dla każdej warstwy, metodę optymalizacji sieci, współczynnik uczenia oraz liczbę epok uczenia.

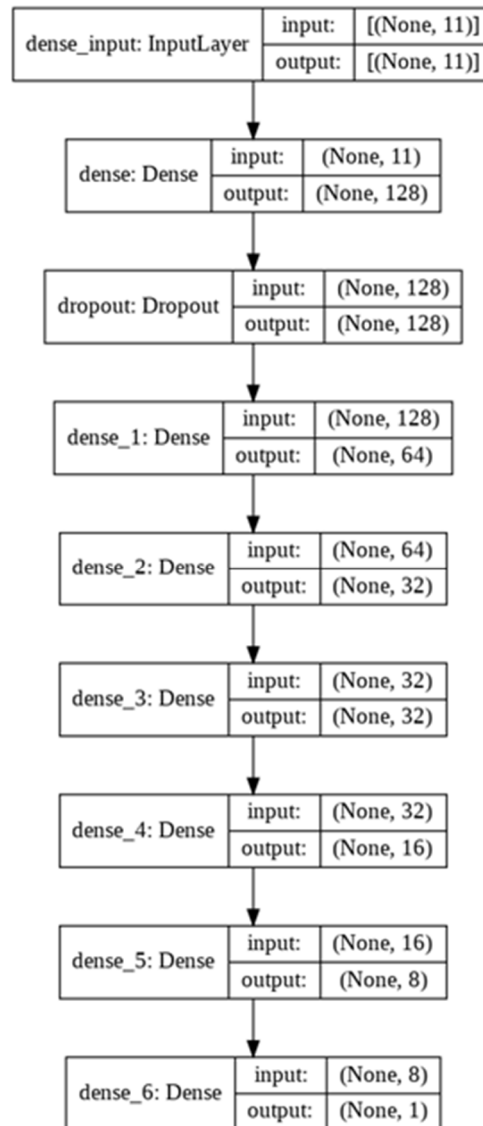
Większość hiperparametrów zostało ustawionych bazując na własnej wiedzy oraz powtarzając uczenie modelu dla różnych zestawów parametrów w celu znalezienia najlepiej działającej sieci neuronowej typu MLP. Hiperparametrami określa się ustawienia konfiguracyjne wpływające na wydajność sieci neuronowej.

Ostatecznymi danymi wejściowymi, które były wykorzystane do predykcji zanieczyszczeń powietrza atmosferycznego NO<sub>x</sub> i SO<sub>2</sub> są tlenek węgla (CO), ozon (O<sub>3</sub>), benzen (C<sub>6</sub>H<sub>6</sub>), pył zawieszony PM10, pył zawieszony PM2.5, ciśnienie atmosferyczne, wilgotność względna powietrza, temperatura powietrza, kierunek wiatru, prędkość wiatru oraz opady atmosferyczne.

Dla porównania wyników przewidywanych zanieczyszczeń została wykorzystana taka sama architektura modelu neuronowego dla predykcji obu wybranych związków. Dodatkowo z uwagi na dużą liczbę wartości w zbiorze danych, do większości badań nad modelami posłużono się jedynie danymi z 2016 roku. Dane te posiadają najmniej sztucznie uzupełnianych wartości. Analiza będzie się składać z oceny modelu dla



zbioru testowego, porównania modelu sztucznej sieci neuronowej z różnymi metodami uczenia maszynowego, oceny efektywności metod skalowania danych wejściowych, porównania funkcji strat wykorzystywanych przy regresji oraz sprawdzenia wpływu wielkości zbioru danych na proces uczenia modelu. Graficzne przedstawienie zaprojektowanego perceptronu wielowarstwowego pokazano na rysunku 3.



Rysunek 3. Zaprojektowany perceptron wielowarstwowy

Przedstawiona architektura zawiera 11 neuronów w warstwie wejściowej, czyli tyle ile zmiennych miało być użytych do przewidywania zanieczyszczeń powietrza atmosferycznego. Każda z warstw ukrytych posiada funkcję aktywacyjną, jaką jest

ReLU i określoną liczbę neuronów, która dla projektowanych modeli była potęgą liczby 2. Model posiadał 6 warstw ukrytych, w których w kolejności warstw liczba neuronów wynosiła 128, 64, 32, 32, 16 oraz 8. W modelu słowo Dense oznacza, że każdy neuron warstwy jest połączony z każdym neuronem warstwy następnej. Ostatnia warstwa, zwana warstwą wyjściową, posiada tylko jeden neuron. Wynika to z uwagi na regresyjny charakter modelu, dla którego przewidywano tylko wartości jednej zmiennej.

Rectified Linear Unit to nieliniowa funkcja aktywacyjna używana w sieciach neuronowych, której matematyczne równanie wynosi:

$$y = \max(0, x). \quad (3)$$

ReLU jest funkcją liniową dla wszystkich wartości dodatnich, natomiast przyjmuje wartość zero dla wszystkich ujemnych. Często jest używana zamiast funkcji sigmoidalnej z uwagi na lepsze radzenie sobie z problemem zanikającego gradientu i mniejszą złożoność obliczeniową.

Dodatkowo pomiędzy pierwszą, a drugą warstwą ukrytą występuje funkcja Dropout, która losowo wyłącza z danej warstwy neurony w celu uniknięcia zjawiska nadmiernego dopasowania modelu do danych treningowych. Argumentem w niej zastosowanym jest liczba z zakresu od 0 do 1 przedstawiająca ułamek wyłączonych elementów.

Funkcją straty użytą w modelu był błąd średniokwadratowy MSE, zainicjowanym współczynnikiem uczenia była wartość  $\alpha=0,0005$ , a liczba epok uczenia wynosiła 200.

Wykorzystanym algorytmem optymalizacyjnym był algorytm Adam, który może być stosowany zamiast klasycznego stochastycznego spadku gradientu do iteracyjnego aktualizowania wag połączeń sieci. Nazwa Adam wywodzi się z angielskiego wyrażenia – adaptive moment estimation. Algorytm Adam różni się od algorytmu SGD. W stochastycznym spadku gradientu współczynnik uczenia jest taki sam dla wszystkich aktualizowanych wag i nie zmienia się podczas procesu uczenia. W algorytmie Adam współczynnik uczenia jest utrzymywany dla każdej wagi sieci i jest oddzielnie dostosowywany w miarę rozwoju uczenia modelu. Algorytm Adam można określić, jako metodę wykorzystującą korzyści dwóch innych modyfikacji stochastycznego spadku gradientu. Pierwszą z nich jest Adaptive Gradient Algorithm, czyli AdaGrad, która zarządza współczynnikiem uczenia dla każdego parametru tak, aby poprawić wydajność. Wykorzystywany w problemie rzadkiego gradientu (z ang. sparse gradient) spowodowanego za słabym sygnałem wejściowym danych treningowych żeby uczyć sieć w sposób efektywny. Drugą metodą jest RMSProp, czyli Root Mean Square Propagation. Wykorzystuje on współczynnik uczenia adaptacyjnego wyznaczony na podstawie średnich ostatnich wielkości gradientów wag dla poszczególnych parametrów. Oznacza to, że algorytm dobrze radzi sobie z bieżącymi i zmiennymi problemami podczas uczenia, na przykład szumem. Algorytm Adam zamiast dostosowywać współczynniki uczenia na podstawie średniej tak jak ma to miejsce w algorytmie RMSProp, używa również niecentrowanej wariancji. Inaczej mówiąc, algorytm liczy eksponencjalnie przemieszczającą się średnią gradientu i kwadratu gradientu. Parametry  $\beta_1$  i  $\beta_2$  kontrolują tempo zmniejszania się tych średnich. Zazwyczaj na początku  $\beta_1$  i  $\beta_2$  ma wartość bliską 1, co powoduje, że odchylenie momentu ma szacowaną wartość 0. Algorytm Adam jest jednym z najchętniej wykorzystywanych algorytmów o stosunkowo prostej

konfiguracji, gdzie domyślne ustawienia parametrów radzą sobie dobrze z większością problemów.

## 5. Analiza jakości modelu MLP dla predykcji zanieczyszczeń powietrza atmosferycznego

Podczas procesu uczenia sztucznych sieci neuronowych należy pamiętać, że pomimo takiej samej struktury modeli dla przewidywanych zanieczyszczeń powietrza atmosferycznego należy stworzyć dwa oddzielne modele i traktować to jak dwa oddzielne problemy. Każda sieć neuronowa przewiduje inne wartości i korzystanie z jednego modelu doprowadzi do błędnych wyników.

Do oceny działania sztucznych sieci neuronowych można wykorzystać wyznaczone wartości MAE, RMSE,  $R^2$ .

MAE (z ang. Mean Absolute Error) jest to średnia arytmetyczna wartość bezwzględna różnicy pomiędzy wartością prawdziwą, a przewidywaną. Opisuje błąd miarowy, którym jest obciążona predykcja. Oprócz metryki oceny modelu może być wykorzystany, jako funkcja straty w sieciach neuronowych dla zadań regresji.

RMSE (z ang. Root Mean Square Error) jest to pierwiastek błędu średniokwadratowego. Jest to metryka opisująca odchylenie standardowe błędu predykcji.

$R^2$  (z ang. R squared) jest to miara jakości dopasowania modelu z zakresu [0,1], nazywana współczynnikiem determinacji. Opisuje na ile model wyjaśnia zmienną zależną. W przypadku więcej niż jednego predyktora należy skorzystać ze skorygowanego  $R^2$ . Dla badania wyników wartości skorygowanego  $R^2$  dla przedstawionego problemu została sprawdzona, lecz nie różniła się znacząco od wartości  $R^2$ .

W tabeli 3 zestawiono wyliczone dla modelu neuronowego wartości MAE, RMSE oraz  $R^2$ .

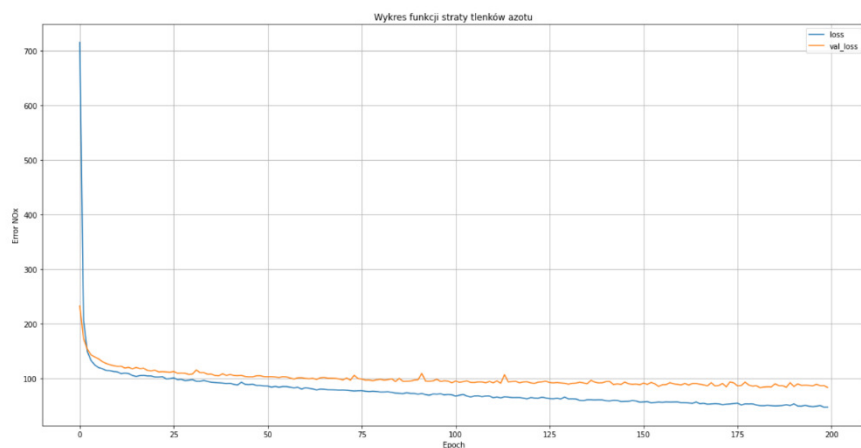
Tabela 3. Wskaźniki oceny modelu neuronowego

	NO <sub>x</sub>	SO <sub>2</sub>
MAE	6,260377	4,487782
RMSE	9,085141	6,687186
$R^2$	0,812890	0,839712

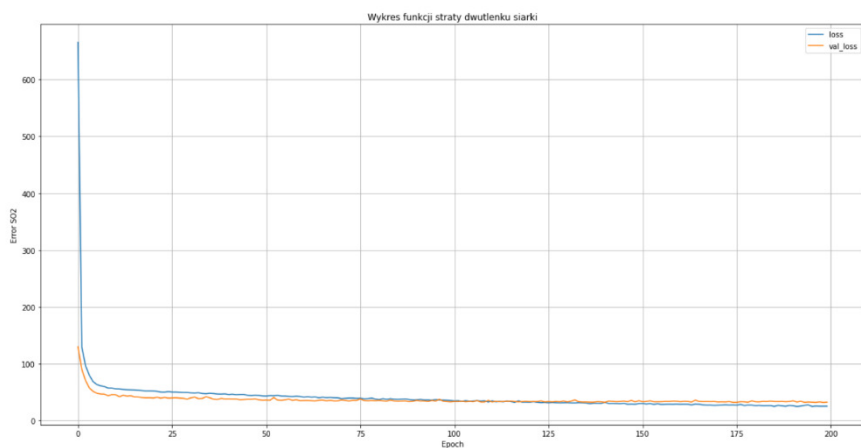
Na rysunkach 4 i 5 pokazano wykresy funkcji straty, natomiast na rysunkach 6 i 7 relacja między rzeczywistymi a przewidywanymi wartościami zanieczyszczenia powietrza NO<sub>x</sub> i SO<sub>2</sub>.

Porównując wykresy funkcji straty, wartości metryk oceny poszczególnych modeli dla zbioru testowego oraz to jak wygląda relacja między wartościami rzeczywistymi, a przewidywanymi przedstawionych zanieczyszczeń powietrza można stwierdzić, że dane dotyczące dwutlenku siarki są łatwiejsze do zinterpretowania dla przedstawionego modelu niż tlenku azotu. Jednym z powodów takiego stanu rzeczy może być mniejsze odchylenie standardowe danych dla SO<sub>2</sub>. W przypadku funkcji straty nie zauważono efektu przeuczenia ani niedouczenia sieci. Wartości MAE i RMSE są dostatecznie małe by można mówić o dobrej ocenie jakości modelu,

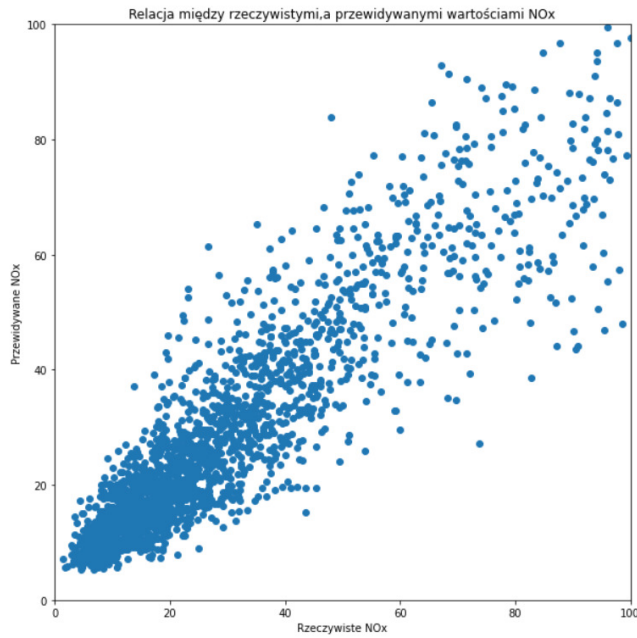
a wartość  $R^2$  (powyżej 0.8) jest satysfakcjonującym wynikiem dla tak dużego zbioru danych. Z relacji między wartościami rzeczywistymi i przewidywanymi można stwierdzić, że udało się przewidzieć wartości na tyle, aby zauważyć pewną zależność. Jeżeli wartości na wykresach są zbliżone do linii przekątnej przechodzącej przez początek układu współrzędnych to relacja wykazuje pewną liniowość, czyli regresja została dobrze zamodelowana. W przypadku jeżeli punkty na wykresach są oddalone od przekątnej i rozrzucone po całej przestrzeni tak, że nie można zauważyć żadnej relacji, oznacza to, że nie udało się dobrze zamodelować regresji.



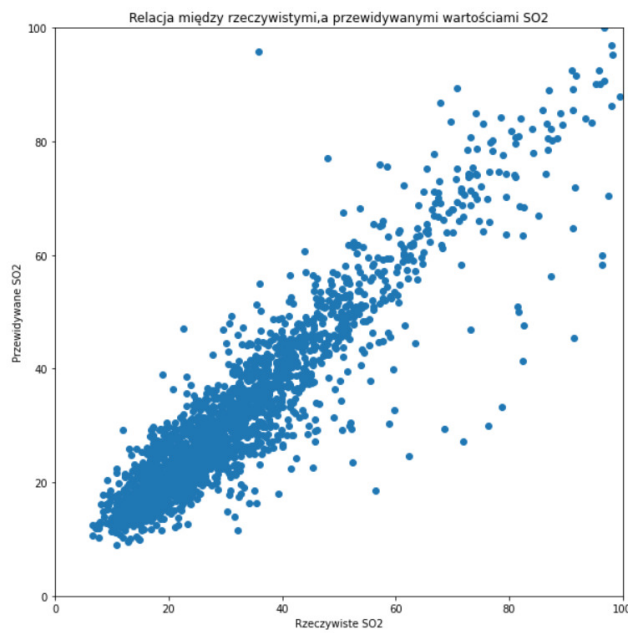
Rysunek 4. Wykres funkcji straty dla modelu neuronowego do predykcji  $NO_x$  przy wyborze różnej funkcji straty



Rysunek 5. Wykres funkcji straty dla modelu neuronowego do predykcji  $SO_2$  przy wyborze różnej funkcji straty



Rysunek 6. Relacja między rzeczywistymi a przewidywanymi wartościami zanieczyszczenia powietrza NO<sub>x</sub> przy wyborze różnej funkcji straty



Rysunek 7. Relacja między rzeczywistymi a przewidywanymi wartościami zanieczyszczenia powietrza SO<sub>2</sub> przy wyborze różnej funkcji straty

W kolejnym etapie badano wpływ funkcji straty na działanie sieci neuronowej MLP. W tej części badań sprawdzano, czy zmiana funkcji straty na MAE (średni błąd bezwzględny), zwiększy wydajność modelu sieci neuronowej. Pozostałe hiperparametry modeli pozostały bez zmian.

Tabela 4. Metryki oceny modeli neuronowych przy wyborze różnej funkcji straty

	Metryki oceny modeli z funkcją straty MAE		Metryki oceny modeli z funkcją straty MSE	
	NO <sub>x</sub>	SO <sub>2</sub>	NO <sub>x</sub>	SO <sub>2</sub>
MAE	6,484975	4,334850	6,260377	4,487782
RMSE	9,5697767	6,649520	9,085141	6,687186
R <sup>2</sup>	0,792395	0,841512	0,812890	0,839712

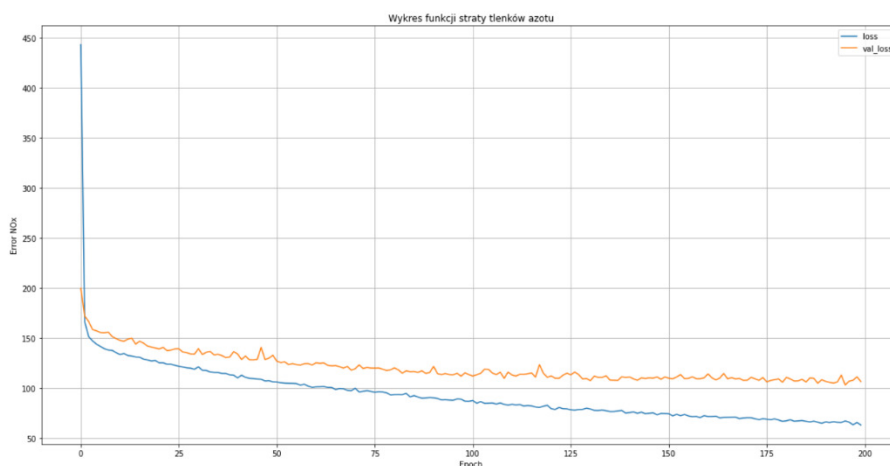
Porównując obydwie funkcje straty można zauważyć, że dla MAE model SO<sub>2</sub> wypadł trochę lepiej, natomiast NO<sub>x</sub> gorzej w porównaniu do MSE. Jednym z możliwych powodów takiego stanu jest problem średniego błędu bezwzględnego z radzeniem sobie z dużymi wartościami błędów, co przekłada się na gorszą wydajność. Średni błąd kwadratowy penalizuje duże wartości błędów, co przekłada się na lepszą wydajność w modelach gdzie błędy są większe. Jednak różnice w wartościach nie są aż tak znaczące, aby można było stwierdzić, że zmiana MSE na MAE jest dobrym rozwiązaniem.

Kolejnym etapem badań była ocena efektywności metod skalowania danych wejściowych na podstawie wydajności modelu sztucznej sieci neuronowej (tabela 5). Analizując wszystkie otrzymane wskaźniki można zaobserwować, że najlepsze wyniki otrzymywane były wykorzystując skalowanie danych wejściowych za pomocą standaryzacji Z-score. Wyniki z wykorzystaniem skalowania MinMax były poprawne. Najgorsze wartości i najbardziej odstające od poprawnego wyniku uzyskano z modelu bez wykorzystania normalizacji danych. Dodatkowo należy wspomnieć, że próba przeskalowania zmiennej zależnej dała znacznie gorsze wyniki, dlatego wartości przewidywane zostały pozostawione w pierwotnej skali. Możliwym powodem takiej sytuacji jest duża liczba wartości zmiennej zależnej, która po przeskalowaniu należy do zbyt małego przedziału by model mógł odróżniać poszczególne wartości. Podsumowując, porównanie różnych technik pokazało, że należy skalować dane w celu wydajniejszego i szybszego procesu uczenia.

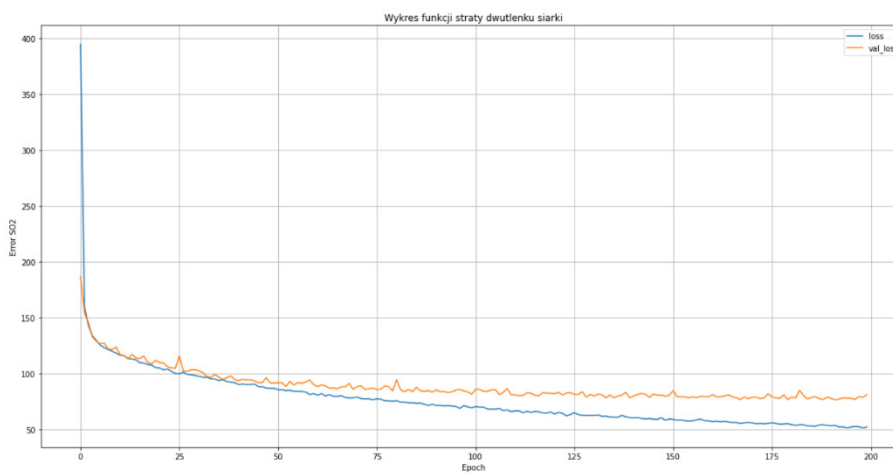
Tabela 5. Metryki oceny modeli neuronowych przy różnym przygotowaniu danych

	Metryki oceny modeli ze standaryzacją Z-score		Metryki oceny modeli z normalizacją skali MinMax	
	NO <sub>x</sub>	SO <sub>2</sub>	NO <sub>x</sub>	SO <sub>2</sub>
MAE	6,260377	4,487782	6,960954	4,902402
RMSE	9,085141	6,687186	9,815684	7,014597
R <sup>2</sup>	0,812890	0,839712	0,781588	0,823632
	Metryki oceny modeli bez normalizacji i standaryzacji			
	NO <sub>x</sub>	SO <sub>2</sub>		
MAE	8,312647	5,679094		
RMSE	11,911365	8,295524		
R <sup>2</sup>	0,678369	0,753338		

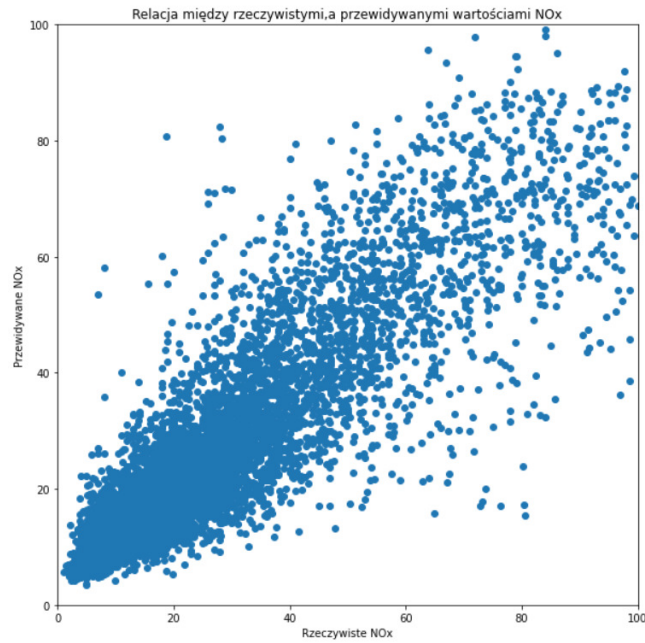
W kolejnym etapie badań analizowano wpływ wielkości zbioru danych na proces uczenia modelu neuronowego. W pierwszej części eksperymentu wykorzystywano dane z 2016 roku, gdzie liczba rekordów wynosiła 7225. W drugiej części eksperymentu użyto danych z okresu 2015-2017, co stanowiło 21657 rekordów. Na rysunkach 8 i 9 pokazano wykresy funkcji straty, natomiast na rysunkach 10 i 11 relacja między rzeczywistymi a przewidywanymi wartościami zanieczyszczenia powietrza  $\text{NO}_x$  i  $\text{SO}_2$  przypadku uczenia modeli neuronowych danymi pochodzącymi z 3 kolejnych lat.



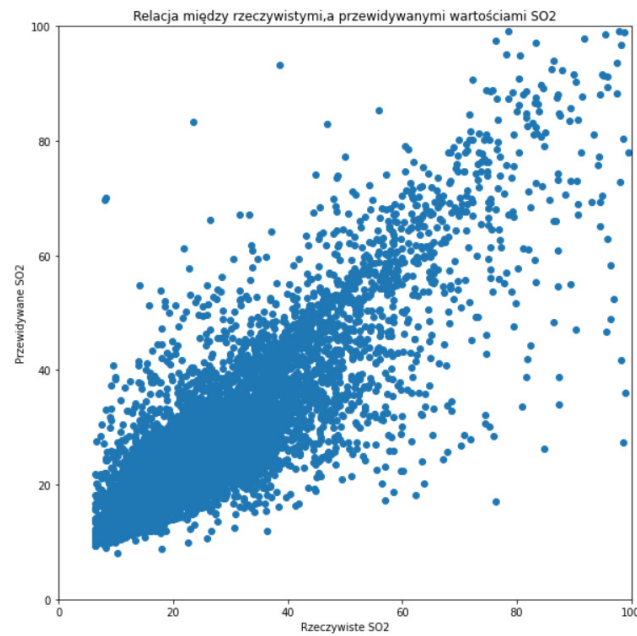
Rysunek 8. Wykres funkcji straty dla modelu neuronowego do predykcji  $\text{NO}_x$  przy różnym przygotowaniu danych



Rysunek 9. Wykres funkcji straty dla modelu neuronowego do predykcji  $\text{SO}_2$  przy różnym przygotowaniu danych



Rysunek 10. Relacja między rzeczywistymi a przewidywanymi wartościami zanieczyszczenia powietrza NO<sub>x</sub> przy różnym przygotowaniu danych



Rysunek 11. Relacja między rzeczywistymi a przewidywanymi wartościami zanieczyszczenia powietrza SO<sub>2</sub> przy różnym przygotowaniu danych



W tabeli 6 przedstawiono wskaźniki oceny modelu neuronowego uczonego na danych pochodzących z 3 kolejnych lat.

Tabela 6. Wskaźniki oceny modelu neuronowego uczonego na danych pochodzących z 3 kolejnych lat

	NO <sub>x</sub>	SO <sub>2</sub>
MAE	6,955287	6,270566
RMSE	10,160615	9,212786
R <sup>2</sup>	0,752853	0,673825

Porównując otrzymane wartości wskaźników i wykresy z analizą jakości modelu neuronowego MLP dla predykcji zanieczyszczeń z 2016 roku można stwierdzić, że większa liczba danych uczących wymaga innej architektury sieci. Każdy model powinien być dostosowywany do określonego zadania z uwzględnieniem rozmiaru danych. Porównując modele dla danych z 2016 roku i dla danych z okresu 2015-2017 można powiedzieć, że dla większego zbioru, mimo iż wyniki wskazują gorsze wartości to przy tylu danych model poradził sobie relatywnie dobrze. Sieć neuronowa wykorzystująca zbiór walidacyjny, w obu przypadkach po kilkunastu epokach przestaje poprawiać swój wynik. Mimo to nie następuje przeuczenie i gwałtowny wzrost błędu, co może świadczyć o możliwości dalszego rozwijania architektury modelu.

## 6. Podsumowanie

W dzisiejszych czasach każdy stara się podążać za trendami, a jednym z nich jest bycie ekologicznym. Ochrona Ziemi powinna być nie tylko modna, ale też głęboko zakorzenionym w podświadomości obowiązkiem. W ten trend idealnie wpisują się przeprowadzone badania.

W artykule przedstawiono wyniki próby predykcji wartości tlenków azotu i dwutlenku siarki wykorzystując do tego celu metodę sztucznej inteligencji w postaci perceptronu wielowarstwowego.

Modele sztucznej sieci neuronowej typu MLP wykorzystujące dane zarejestrowane w jednej z dolnośląskich stacji pomiarowych działały poprawnie, jednak nadal istnieje duże pole do rozwoju i rozbudowania istniejącej koncepcji. Analiza modelu regresji pod kątem poszczególnych parametrów sieci pozwala lepiej zrozumieć sposób działania sztucznych sieci neuronowych. Porównywanie różnych przypadków, na przykład modeli ze znormalizowanymi i bez normalizacji danymi, pokazały jak ważny jest każdy aspekt tworzenia sieci. Możliwość przewidywania konkretnych zanieczyszczeń powietrza atmosferycznego na podstawie czynników pogodowych i innymi mierzonych parametrów z wykorzystaniem sieci neuronowych typu MLP może mieć realne zastosowanie w przypadku braku odpowiedniej aparatury do pomiarów zanieczyszczeń, jak również przy uzupełnianiu niekompletnych danych pomiarowych, czy też kontrolowaniu dopuszczalnego poziomu zanieczyszczeń.

**LITERATURA**

1. BATTITI R.: First and second order methods for learning: Between steepest descent and Newton's method. *Neural Computation*, **4**(1992)2, p. 141-166.
2. BOTTOU L., BOUSQUET O.: The Tradeoffs of Large Scale Learning. In: SRA S., NOWOZIN S., WRIGHT S.J. (eds.), *Optimization for Machine Learning*. MIT Press, Cambridge 2012, p. 351-368.
3. DUCH W., KORBICZ J., RUTKOWSKI L., TADEUSIEWICZ R. (red.): *Sieci neuronowe*. W: NAŁĘCZ M. (red.), *Biocybernetyka i inżynieria biomedyczna 2000*, tom VI, Akademicka Oficyna Wydawnicza EXIT, Warszawa 2000.
4. FORESEE F.D., Hagan M.T.: Gauss-Newton approximation to Bayesian learning. *Proceedings of the International Joint Conference on Neural Networks*, June, 1997.
5. GOODFELLOW I., BENGIO Y., COURVILLE A.: *Deep Learning*. MIT Press, 2016.
6. KINGMA D.P., BA J.A.: A method for stochastic optimization. Conference paper at ICLR 2015.
7. KORBICZ J., OBUCHOWICZ A., UCIŃSKI D.: *Sztuczne sieci neuronowe - podstawy i zastosowania*. Akademicka Oficyna Wydawnicza PLJ, Warszawa 1994.
8. MACKAY D.J.C.: Bayesian interpolation. *Neural Computation*, **4**(1992)3, p. 415-447.
9. MOLLER M.F.: A scaled conjugate gradient algorithm for fast supervised learning. *Neural Networks*, **6**(1993), p. 525-533.
10. NIELSEN M.A.: *How the backpropagation algorithm works*. Neural Networks and Deep Learning. Determination Press, 2015.
11. OSOWSKI S.: *Sieci neuronowe do przetwarzania informacji*. Oficyna Wydawnicza Politechniki Warszawskiej, Warszawa 2020.
12. RASCHKA S.: *Python machine learning*. Packt Publishing Ltd. 2015.
13. SIEDLECKI J. (red.): *Wykorzystanie sztucznych sieci neuronowych w modelowaniu ekonomicznym*. Wydawnictwo Akademii Ekonomicznej im. Oskara Langego we Wrocławiu, Wrocław 2001.
14. TADEUSIEWICZ R.: *Sieci neuronowe*. Akademicka Oficyna Wydawnicza RM, Warszawa 1993.

Badania zrealizowano w ramach Project Base Learning sfinansowanego z Funduszu Europejskiego w ramach projektu Politechnika Śląska jako Centrum Nowoczesnego Kształcenia opartego o badania i innowacje POWR-03.05.00-00-Z098/17-00.