UNIWERSYTET ŚLĄSKI W KATOWICACH
WYDZIAŁ PRAWA I ADMINISTRACJI

MACIEJ MARCINOWSKI-PRAŻMOWSKI

ASPEKTY PRAKTYCZNE I METODOLOGICZNE ZASTOSOWANIA SZTUCZNYCH SIECI NEURONOWYCH W BADANIACH KRYMINALISTYCZNYCH NA PRZYKŁADZIE BADAŃ PISMOZNAWCZYCH

ROZPRAWA DOKTORSKA

PROMOTOR:
DR HAB. MAREK LEŚNIAK, PROF. UŚ

KATOWICE 2023

Spis treści

Wprowadzenie

Przedmiotem niniejszej rozprawy są wybrane problemy praktyczne i metodologiczne zastosowania sztucznych sieci neuronowych do badań kryminalistycznych. Zdaniem autora są to problemy istotne, ponieważ dziedzina kryminalistyki obliczeniowej bardzo szybko się rozwija, a dopiero ich rozwiązanie umożliwiać będzie określenie zgodnych z potrzebami praktyki kierunków tego rozwoju i zastosowań sieci neuronowych w kryminalistyce. Ponieważ sieci neuronowe są nieinterpretowalne, a w związku z tym trudne w ewaluacji, przy czym osiągają na ogół bardzo wysoką trafność, niską rzetelność i są podatne na manipulacje, to propozycje ich szerszego zastosowania w kryminalistyce rodzą wątpliwości, dopóki tych problemów nie uda się rozwiązać. Sztuczne sieci neuronowe służyć też mogą tzw. antykryminalistyce[1] (są to na przykład działania mające na celu ukrycie czynu zabronionego lub jego sprawcy), przede wszystkim do fałszowania danych w wielkich ilościach, wobec czego określić należy, czy metody zajmujące się badaniem takich danych będą skuteczne wobec takich fałszerstw (*e.g.* badania pismoznawcze wobec podpisów fałszowanych za pomocą sieci neuronowych). Autor zdecydował oprzeć się na przykładzie badań pismoznawczych, ponieważ kryminalistyka jest dziedziną szeroką, a udzielenie odpowiedzi dla stawianych pytań wymaga jednolitej podstawy teoretycznej, u przeprowadzenia porównywalnych eksperymentów. Decyzja ta uzasadniona jest dwoma argumentami, otóż: i) problematyka badań pismoznawczych jest wysoce wizualna (więc umożliwia wizualizacje schematów decyzyjnych sieci neuronowych); ii) oraz wysoce abstrakcyjna (nie można poprzestać na prostych i potencjalnie fałszywych wyjaśnieniach wynikających z wizualizacji).

Celem autora było udzielenie odpowiedzi na następujące pytania badawcze:

I) Jak dokonywać ewaluacji sztucznych sieci neuronowych dla potrzeb kryminalistyki?

II) Jak dokonywać interpretacji sztucznych sieci neuronowych dla potrzeb kryminalistyki?

III) Jak wykrywać fałszerstwa popełniane z wykorzystaniem sztucznych sieci neuronowych?

1 P. Chlebowicz, P. Łabuz, T. Safjański, *Antykryminalistyka. Taktyka i technika działań kontrwykrywczych*, Warszawa 2022.

Adekwatnie do postawionych pytań badawczych, autor przeprowadził zaprojektowane przez siebie eksperymenty, opracowując: i) sztuczne sieci neuronowe do weryfikacji wykonawstwa rękopisów (*i.e*. które przypisują parę rękopisów do klasy „ten sam wykonawca" lub „różni wykonawcy"), celem przedstawienia problematyki ewaluacji sieci neuronowych; ii) sztuczne sieci neuronowe do identyfikacji wykonawców rękopisów (*i.e*. które przypisują dany rękopis do danego wykonawcy należącego do skończonego zbioru możliwych wykonawców), celem przedstawienia problematyki interpretacji sieci neuronowych; iii) sztuczną sieć neuronową do generowania fałszywych wariantów podpisów określonych osób, celem przedstawienia problematyki wykrywania fałszerstw popełnianych za pomocą sieci neuronowych. Ze względu na eksploracyjny charakter pracy, hipotezy formułowane były w kontekście poszczególnych badań empirycznych, które wymieniono powyżej. Przeprowadzone eksperymenty szczegółowo i instruktażowo udokumentowano w ogólnie dostępnych repozytoriach, starając się w ten sposób zapewnić transparentność i reprodukowalność prowadzonych badań.

Adekwatnie do charakteru opisanych w dysertacji badań, autor podzielił rozprawę na dwie części. W części pierwszej autor stworzył tło teoretyczne badań własnych, gdzie opisał: i) istotę sztucznych sieci neuronowych; ii) podstawy kryminalistyki obliczeniowej; iii) zastosowania sztucznych sieci neuronowych w kryminalistyce i antykryminalistyce; iv) aspekty prawne zastosowania sztucznych sieci neuronowych w kryminalistyce (ze szczególnym uwzględnieniem projektu Aktu w sprawie sztucznej inteligencji[2]); v) aspekty praktyczne zastosowania sztucznych sieci neuronowych w kryminalistyce; iv) aspekty metodologiczne zastosowania sztucznych sieci neuronowych w kryminalistyce. W części drugiej autor opisał przeprowadzone przez siebie eksperymenty, stanowiące przykład: i) ewaluacji sztucznych sieci neuronowych dla potrzeb kryminalistyki, dowodząc konieczności ewaluacji sieci neuronowych poprzez falsyfikacyjne testowanie ich rzetelności; ii) interpretacji sztucznych sieci neuronowych dla potrzeb kryminalistyki, dowodząc możliwości tworzenia na potrzeby kryminalistyki interpretowalnych sieci neuronowych; iii) fałszowania podpisów za pomocą sztucznej sieci neuronowej, wobec której badano

2 Wniosek dotyczący rozporządzenia Parlamentu Europejskiego i Rady (UE) nr COM/2021/206 final z dnia 21 kwietnia 2021 r. ustanawiającego zharmonizowane przepisy dotyczące sztucznej inteligencji (akt w sprawie sztucznej inteligencji) i zmieniającego niektóre akty ustawodawcze unii.

sposoby wykrywania takich fałszerstw metodami pismoznawczymi, wykazując dotąd niski stopień zagrożenia i niewielką przydatność takich metod fałszowania; iv) na koniec autor przeprowadził całościową dyskusję uzyskanych wyników w kontekście przyszłości sztucznych sieci neuronowych w kryminalistyce i antykryminalistyce.

Część teoretyczna

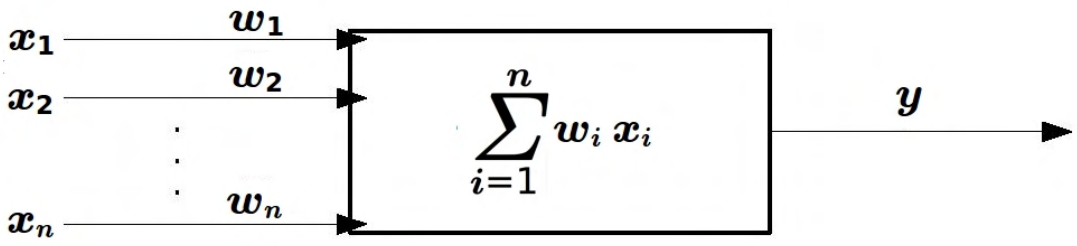
Rozdział 1. Sztuczne sieci neuronowe.

1.1. Wprowadzenie. Sztuczne sieci neuronowe (*artificial neural networks,* ANN) stanowią dominującą metodę uczenia maszynowego (*machine learning*, ML), a to najważniejszą dziedzinę sztucznej inteligencji (*artificial intelligence*, AI). Większość stanowią dzisiaj głębokie sieci neuronowe (*deep neural networks, i.e.* składające się z co najmniej trzech warstw neuronów nieliniowych), które obejmowane są określeniem uczenie głębokie (*deep learning*).

Przede wszystkim sztuczne sieci neuronowe są funkcjami matematycznymi. Funkcja jest zaś relacją na dwóch zbiorach, która przekształca zbiór argumentów na zbiór wartości. W przypadku sieci neuronowych: i) zbiór argumentów zawiera numerycznie zakodowane dane, które dany problem opisują; ii) a zbiór wartości zawiera numerycznie zakodowane rozwiązania tego problemu. Rozwiązanie problemu sprowadza się więc do do przekształcenia jednego zbioru w drugi, czyli do przeliczenia opisów problemu na jego rozwiązania. Problem polega na odnalezieniu funkcji, która tego dokona. Mając więc zbiór danych wejściowych i wyjściowych, czyli dane uczące, można zastosować sztuczną sieć neuronową, która w procesie uczenia przybliży się do funkcji rozwiązującej dany problem. Sztuczna sieć neuronowa jest wysoce zorganizowaną strukturą prostych funkcji matematycznych (sekwencji iloczynów, sum i nieliniowości, nazywanych neuronami), połączonych są ze sobą za pomocą modyfikowalnych parametrów (wag numerycznych, nazywanych synapsami), których automatycznie wyliczane korekty składają się na proces uczenia sieci. Proces ten ma na celu minimalizację błędów wyliczanych za pomocą funkcji kosztu, która ocenia na ile predykcje modelu odbiegają od oczekiwanych rozwiązań.

1.2. Budowa sztucznej sieci neuronowej.

Neuron liniowy. Typowy neuron sztuczny posiada wiele wejść i jedno wyjście, a składa się z wag synaptycznych, sumatora i danej nieliniowej funkcja aktywacji. W przypadku neuronów liniowych (rys. 1.2.1) funkcja ta nie występuje, przesądzając o ich liniowym sposobie przetwarzania danych.



Rysunek 1.2.1. Schemat neuronu liniowego.

Źródło: opracowanie własne.

Wagi synaptyczne w postaci współczynnika w_i przyjmować mogą wartości z dowolnie wyznaczonych przedziałów, a są na ogół losowane z przedziału $[0, 1]$, zaś sygnały wejściowe x_i są do niego skalowane. Sztuczny neuron można ująć jako funkcję:

$$y = w_1x_1 + w_2x_2 + \cdots + w_nx_n = \sum_{i=1}^{n} w_ix_i \qquad \text{(Równanie 1.2.1)}$$

Gdzie:

x_i – element sygnału wejściowego;

w_i – dana waga synaptyczna;

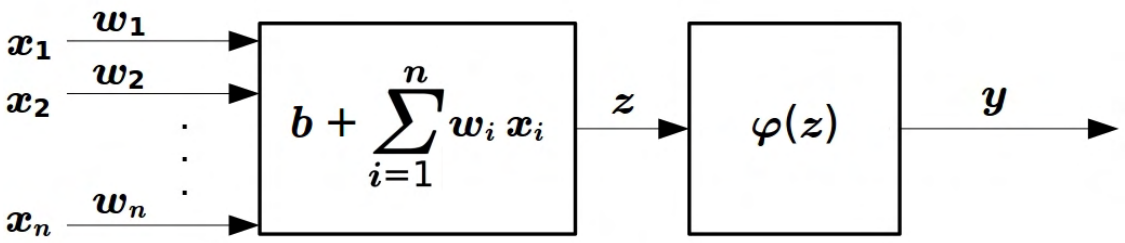
y – sygnał wyjściowy;

$\sum_{i=1}^{n}$ – suma elementów od $i = 1$ do n.

Sygnał wyjściowy neuronu jest więc sumą – iloczynów wag i sygnałów wejściowych. Liniowość takiego neuronu ująć można tutaj intuicyjnie, jako prostą proporcjonalność sygnału wyjściowego do sygnałów wejściowych. Proporcjonalność prosta oznacza, że stosunek wartości wyjściowej do argumentu wejściowego jest taki, iż iloraz tych wartości jest stałym współczynnikiem – w omawianym przypadku jest to wartość synapsy.

Neuron nieliniowy. Cechą determinującą nieliniowy sposób przetwarzania informacji przez sztuczny neuron (a dzięki temu przez samą sieć) jest wprowadzany do jego konstrukcji element nieliniowy, na ogół pod postacią danej nieliniowej funkcji aktywacji. Pierwszym tego typu neuronem była zarazem pierwsza

sieć neuronowa – *Perceptron* – zbudowana w 1957 roku przez F. Rosenblatta w *Cornell Aeronautical Laboratory.* Analiza właściwości takiego rozwiązania poruszona zostanie w kontekście liniowości i nieliniowości całych sieci. Zauważyć jednak warto za R. Tadeusiewiczem, że jest to rozwiązanie w naturalny sposób zasadne: „[...] ponieważ rzeczywiste biologiczne neurony są nieliniowe”[3]. Neuron taki przybiera postać następującego schematu (rys. 1.2.2):



Rysunek 1.2.2. Schemat neuronu nieliniowego.

Źródło: opracowanie własne.

Gdzie sygnał z w neuronie liniowym stanowił jego sygnał wyjściowy (*i.e.* y), tutaj jest wartością „pobudzenia” neuronu przez sumę iloczynów wag synaptycznych i sygnałów wejściowych, zwiększoną o wartość ewentualnego modyfikowalnego parametru b (*bias*). Co zdefiniować można jako:

$$z = b + \sum_{i=1}^{n} w_i x_i \qquad \text{Równanie 1.2.2)}$$

Sygnał z jest następnie argumentem danej nieliniowej funkcji aktywacji $\varphi()$, stąd sygnał wyjściowy neuronu y definiowany jest jako:

$$y = \varphi(z) \qquad \text{(Równanie 1.2.3)}$$

Całość zapisać można jako:

$$y = \varphi(b + \sum_{i=1}^{n} w_i x_i) \qquad \text{(Równanie 1.2.4)}$$

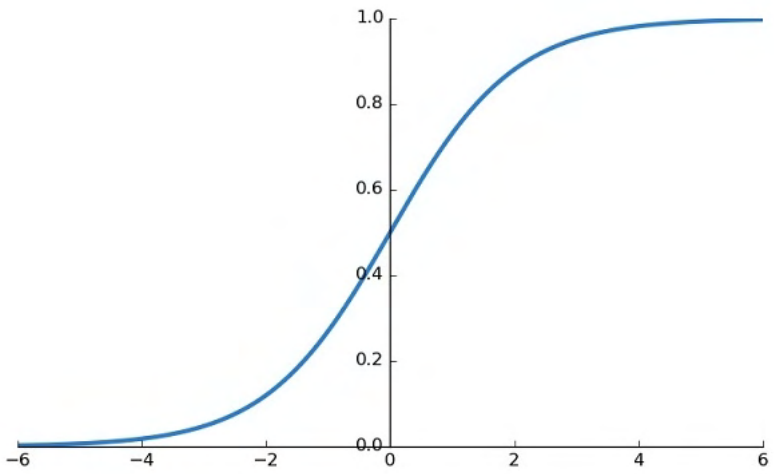
Formy nieliniowości neuronu. Jako przykłady nieliniowości neuronów rozważone będą najpopularniejsze funkcje aktywacji – sigmoidalną, *softmax* i ReLU.

3 R. Tadeusiewicz, *Sieci Neuronowe*, Warszawa 1993, s. 49.

Funkcja sigmoidalna unipolarna skaluje sygnał z do przedziału wartości $[0, 1]$. Standardową jej postać, zdefiniować można jako:

$$\varphi(z) = \frac{1}{1 + e^{-z}} \qquad \text{(Równanie 1.2.5)}$$

Gdzie współczynnik e jest to tak zwana liczba Eulera. Schematycznie funkcja sigmoidalna przedstawia się następująco (rys. 1.2.3):



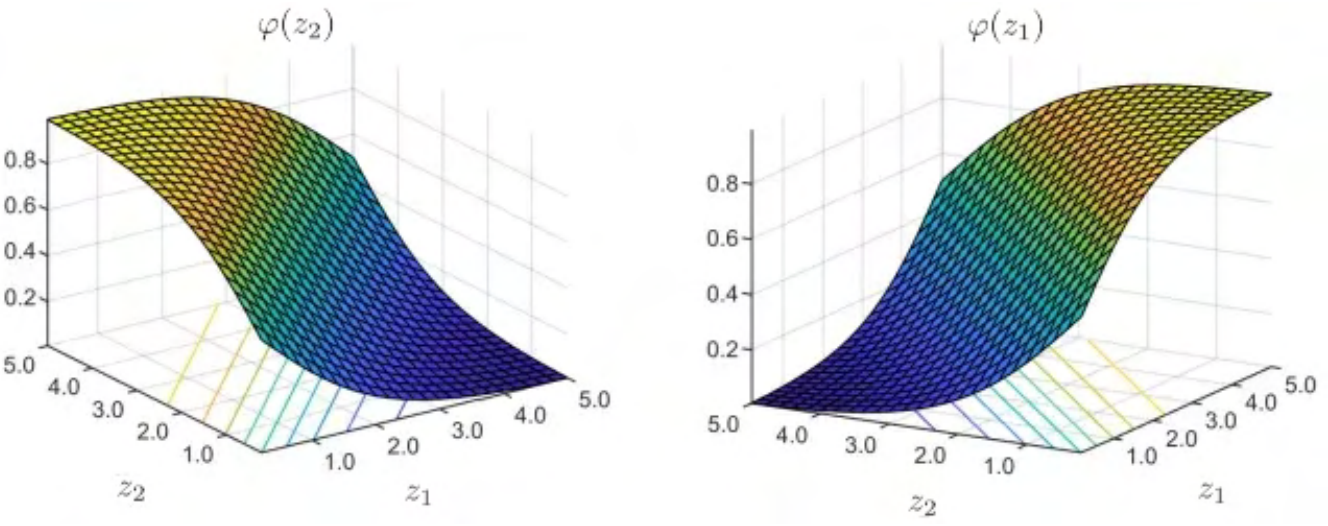
Rysunek 1.2.3. Przykładowy wykres funkcji sigmoidalnej unipolarnej.
Źródło: https://themaverickmeerkat.com/2019-10-23-Softmax/, dostęp 15 listopada 2022 r.

Funkcja sigmoidalna znajduje najczęstsze zastosowanie na neuronach wyjściowych, kiedy oczekiwane odpowiedzi przyjmują wartości $\{0, 1\}$. W odmianie bipolarnej, funkcja sigmoidalna (tangens hiperboliczny) skaluje do przedziału $[-1, 1]$.

Funkcja *softmax* znajduje zastosowanie tylko na neuronach wyjściowych sieci, bowiem równocześnie skaluje sygnały wyjściowe z wielu neuronów do przedziału $[0, 1]$, zapewniając przy tym, że ich suma równa będzie 1. Stąd, mogą one być interpretowane jako prawdopodobieństwa przynależności do poszczególnych klas, gdzie każdy neuron reprezentuje jedną klasę. Funkcja *softmax* dla i-tego neuronu (gdzie $i = j = 1, 2, ..., n$), wobec n-tej liczby klas (gdzie $n > 1$), przybierze postać poniższą:

$$\varphi(z_i) = \frac{e^{z_i}}{\sum_{j=1}^{n} e^{z_j}} \qquad \text{(Równanie 1.2.6)}$$

Wykres funkcji *softmax* dla $n = 2$ przedstawia się następująco (rys. 1.2.4):



Rysunek 1.2.4. Przykładowy wykres funkcji *softmax* dla $n = 2$, gdzie zaobserwować można, iż aktywacje neuronów są od siebie wzajemnie zależne, *i.e.* im wyższa wartość pobudzenia jednego neuronu względem drugiego, tym wyższa aktywacja tego neuronu względem drugiego.

Źródło: https://nhigham.com/2021/01/12/what-is-the-softmax-function/, dostęp 15 listopada 2022 r.

Funkcja ReLU (*Rectifier function* lub *Rectified linear unit*) zaproponowana została w 1975 roku przez K. Fukushimę[4] i stanowi dzisiaj najpopularniejszą funkcję aktywacji na warstwach ukrytych (*i.e.* poprzedzających warstwę wyjściową)[5]. Zdefiniować można ją następująco:

$$\varphi(z) = \begin{cases} z, & \text{jeżeli } z \geq 0 \\ 0, & \text{jeżeli } z < 0 \end{cases} \qquad \text{(Równanie 1.2.7)}$$

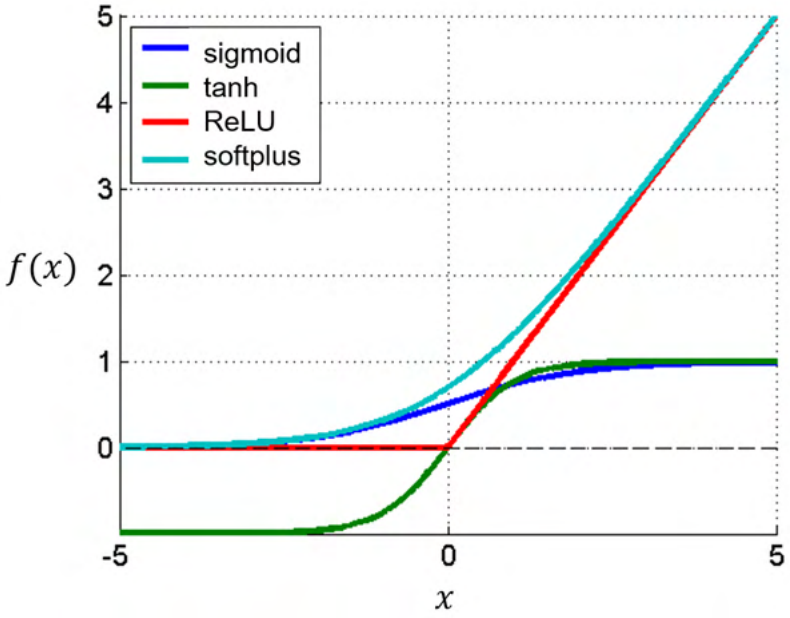
Na prezentowanym poniżej rysunku (rys. 1.2.5), ReLU oznaczona jest kolorem czerwonym. Funkcja *softplus* oznaczona jest kolorem błękitnym – w pobieżnym ujęciu jest ona wygładzoną (*smooth*) wersją funkcji ReLU (czyli różniczkowalną w każdym punkcie, także gdy $z = 0$)[6]. Natomiast funkcje sigmoidalne – unipolarną (*sigmoid*) i

4 K. Fukushima, *Cognitron: A self-organizing multilayered neural network*, „Biological Cybernetics" t. 20 nr 3 (1975), DOI: 10.1007/BF00342633.

5 P. Ramachandran, B. Zoph, Q.V. Le, *Searching for Activation Functions*, arXiv, 27 października 2017 r., http://arxiv.org/abs/1710.05941, s. 1.

6 R. Penrose, *Droga do Rzeczywistości*, Warszwa 2006, s. 104; P. Ramachandran, B. Zoph, Q.V. Le, *Searching for Activation Functions*, arXiv, 27 października 2017 r., http://arxiv.org/abs/1710.05941, s. 7.

bipolarną (*tanh*) – oznaczono stosownie kolorami granatowym i zielonym. Ponadto, $f(x)$ jest równoznaczna $\varphi(z)$.



Rysunek 1.2.5. Przykładowe wykresy wybranych funkcji aktywacji.

Źródło: M. Musiol, *Speeding up Deep Learning Computational Aspects of Machine Learning*, 6 stycznia 2016 r., https://www.researchgate.net/publication/308414212_Speeding_up_Deep_Learning_Computational_Aspects_of_Machine_Learning.

Sieć liniowa. W przypadku modeli liniowych rozważyć warto jako sieć pojedynczą warstwę neuronów (tzw. *Adaptive Linear Element*, ADALINE)[7]. W ramach niniejszego przykładu, oznacza to, iż rozpatrywana jest dowolna grupa neuronów liniowych, które nie są ze sobą połączone, ale którym zadawane będą te same dane wejściowe. Sygnały wejściowe x_i opisujące dany problem, które są wspólne dla wszystkich neuronów, ująć można jako wektor X (macierz jednokolumnową lub tensor pierwszego stopnia)[8], o wymiarach $[n \times 1]$, gdzie $i = 1, 2, ..., n$:

7 R. Tadeusiewicz, *Sieci Neuronowe*, Warszawa 1993, s. 28–30.

8 Stopniowanie tensorów wynika z ilości wymiarów (współrzędnych), które potrzebne są do ich opisania. Tensorami zerowego stopnia są pojedyncze liczby. Tensorami pierwszego stopnia są wektory (macierze jednokolumnowe / jednowymiarowe), które wymagają współrzędnej i podającej numer wiersza, na którym dany element się znajduje. Tensorami drugiego stopnia są macierze (dwuwymiarowe), które wymagają par współrzędnych ij, określających numer wiersza i kolumny danego elementu. Tensory trzeciego stopnia rozumieć można jako wektory, których elementami są macierze, a które wymagają trzech współrzędnych dla każdego elementu (numer macierzy, oraz numer jej wiersza i kolumny). Tensory czwartego stopnia rozumieć można jako wektory, których elementami są tensory trzeciego stopnia. Te ostatnie są szczególnie istotne w uczeniu sieci neuronowych, ponieważ tensor danych uczących jest na ogół tensorem czwartego stopnia. Na przykładzie zbioru obrazów uczących, są one na ogół opisywane jako $bwhc$, gdzie: i) b oznacza

$$X = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \qquad \text{(Równanie 1.2.8)}$$

Który po transponowaniu T (zamianie kolumn z wierszami), zapisać można jako:

$$X^T = [x_1, x_2, ..., x_n] \qquad \text{(Równanie 1.2.9)}$$

Dla każdego j-tego neuronu, gdzie $j = 1, 2, ..., k$, dany wektor wag W_j^T przyjmie postać:

$$W_j^T = [w_{j1}, w_{j2}, ..., w_{jn}] \qquad \text{(Równanie 1.2.10)}$$

Wektory wag W_j wszystkich j-tych neuronów zapisać można po transpozycji jako wiersze macierzy wag W o wymiarach $[k \times n]$:

$$W^T = \begin{bmatrix} w_{11} & \cdots & w_{1n} \\ \vdots & \ddots & \vdots \\ w_{k1} & \cdots & w_{kn} \end{bmatrix} \qquad \text{(Równanie 1.2.11)}$$

Za pomocą poniższego iloczynu zdefiniować można warstwę neuronów (ADALINE):

$$Y = W^T X \qquad \text{(Równanie 1.2.12)}$$

Która jest iloczynem macierzy wag i wektora danych wejściowych:

$$Y = \begin{bmatrix} w_{11}x_1 + w_{12}x_2 & \cdots & w_{1n}x_n \\ w_{21}x_1 + w_{22}x_2 & \cdots & w_{2n}x_n \\ \vdots & \ddots & \vdots \\ w_{k1}x_1 + w_{k2}x_2 & \cdots & w_{kn}x_n \end{bmatrix} \qquad \text{(Równanie 1.2.13)}$$

Gdzie, sygnały wyjściowe Y z całej warstwy stanowią wektor o wymiarach $[k \times 1]$:

numer obrazu (w danej porcji obrazów wejściowych, ze względu na które sieć będzie teraz korygowana, tzw. *batch*), ii) c oznacza numer kanału barw na danym obrazie (tzw. *channel*); iii) w i h oznaczają numer kolumny i wiersza na którym dany piksel się znajduje (są to odpowiednio szerokość, *i.e. width*, oraz wysokość, *i.e. height* obrazu).

$$Y^T = [y_1, y_2, ..., y_k] \qquad \text{(Równanie 1.2.14)}$$

Dla obliczenia pojedynczego sygnału wyjściowego, każdego j-tego neuronu, zastosować można wzór:

$$y_j = W_j^T X = W_j * X \qquad \text{(Równanie 1.2.15)}$$

W równaniu tym, $*$ oznacza iloczyn skalarny wektorów, który jest tożsamy z iloczynem macierzowym wektora wejściowego i transponowanego wektora wag[9]. Ponieważ iloczyn macierzy jednokolumnowej (wektor o wymiarach $n \times 1$) i jednowierszowej (transponowany wektor o wymiarach $1 \times n$) wynosi 1×1, to równoznaczny jest:

$$y_j = w_{j1}x_1 + w_{j2}x_2 + \cdots + w_{jn}x_n = \sum_{i=1}^{n} w_{ji}x_i \qquad \text{(Równanie 1.2.16)}$$

Zatem metoda działania omawianej sieci sprowadza się do wyliczenia przez nią wektora sygnału wyjściowego Y w odpowiedzi na wektor wejściowy X opisujący dany problem. Jest to efekt procesu, w którym na zadany synapsom w_{ji} wektor sygnałów wejściowych x_i, generowane są sygnały wyjściowe y_j, tworzące wektor Y liczący k odpowiedzi, spośród to których najwyższa wartość reakcji y, uzyskana przez j-ty neuron, oznacza iż z całej warstwy (sieci) to on nauczył się rozpoznawać X danej klasy, a sieć nauczyła się go klasyfikować. Stąd naturalny wniosek, że sieć taka może nauczyć się jedynie k różnych klas, ponieważ złożona jest jedynie z k neuronów. Dla przydatnego działania sieci, jeden tylko neuron udzielać powinien najwyższego sygnału wyjściowego na X należący do danej klasy. Gdyby udzieliły go dwa lub więcej neuronów, to dążyć należy do zmniejszenia sygnału wyjściowego ze wszystkich za wyjątkiem jednego. Analogicznie, gdyby dany neuron udzielił najwyższego sygnału odpowiedzi na więcej niż jedną klasę, konieczną jest modyfikacja wag synaptycznych w taki sposób, aby rozpoznawał tylko jedną klasę. Jest to cel (wyznaczany przez dążenie do maksymalizacji użyteczności sieci) aby nauczyć k neuronów przyporządkowywać dane wejściowe do k klas.

9 I. Bronsztejn, K. Siemiendiajew, G. Musiol, H. Mühlig, *Nowoczesne Kompendium Matematyki*, Warszawa 2017, s. 276.

Rozważania wariantów wielowarstwowych, ze względu na liniowość odwzorowania $X \Rightarrow Y$, ograniczyć wystarczy do przykładu dowodzącego, iż wielowarstwowość takich sieci jest pozbawiona większego znaczenia. Można założyć bowiem sieć m-warstwową, gdzie $m = 1, 2$:

$$Y = W_2^T(W_1^T X) \qquad \text{(Równanie 1.2.17)}$$

Ponieważ iloczyny macierzowe są łączne (asocjacyjne), to:

$$Y = (W_1^T W_2^T)X \qquad \text{(Równanie 1.2.18)}$$

Jeżeli przyjąć, że:

$$W_1^T W_2^T = (W_2 W_1)^T = W^T \qquad \text{(Równanie 1.2.19)}$$

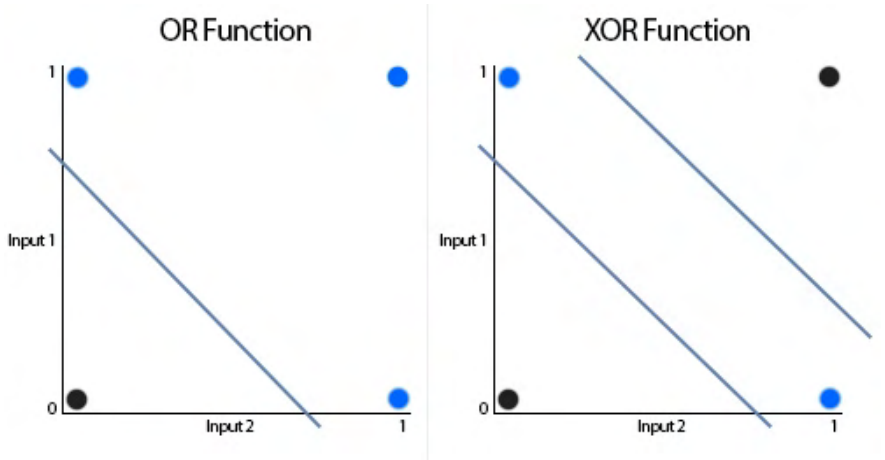
To ponownie:

$$Y = W^T X \qquad \text{(Równanie 1.2.20)}$$

Wynika stąd, iż wielowarstwowość takich sieci – ze względu na ich liniowy sposób odwzorowania – ma marginalne znaczenie[10]. Fakt powyższy przesądza zarazem o ograniczeniach klasyfikacyjnych sieci liniowych, co spowodowało upadek tej dziedziny na przełomie lat 60 i 70 XX wieku (tzw. zima sztucznej inteligencji, *AI winter*)[11].

Posłużyć się tutaj można przykładem graficznym słynnego problemu funkcji XOR (e*xclusive or,* alternatywy rozłącznej), którego perceptron dokonujący odwzorowań liniowych rozwiązać nie może. Na widocznym przykładzie (rys. 1.2.6), dane wejściowe *Input 1* i *Input 2* (innymi słowy x_1 i x_2) przyjmujące wartości $[0, 1]^T$ i $[1, 0]^T$, które spełniają funkcję XOR (jest dla nich prawdziwa), nie mogą zostać wyodrębnione z przestrzeni za pomocą funkcji liniowej, bez jednoczesnego uwzględnienia jednego z przypadków – $[0, 0]^T$ lub $[1, 1]^T$ – dla których funkcja XOR nie jest spełniana. Widoczna na grafice funkcja OR jest alternatywą zwykłą, dla której istnieje rozwiązanie liniowe, którego perceptron może się nauczyć.

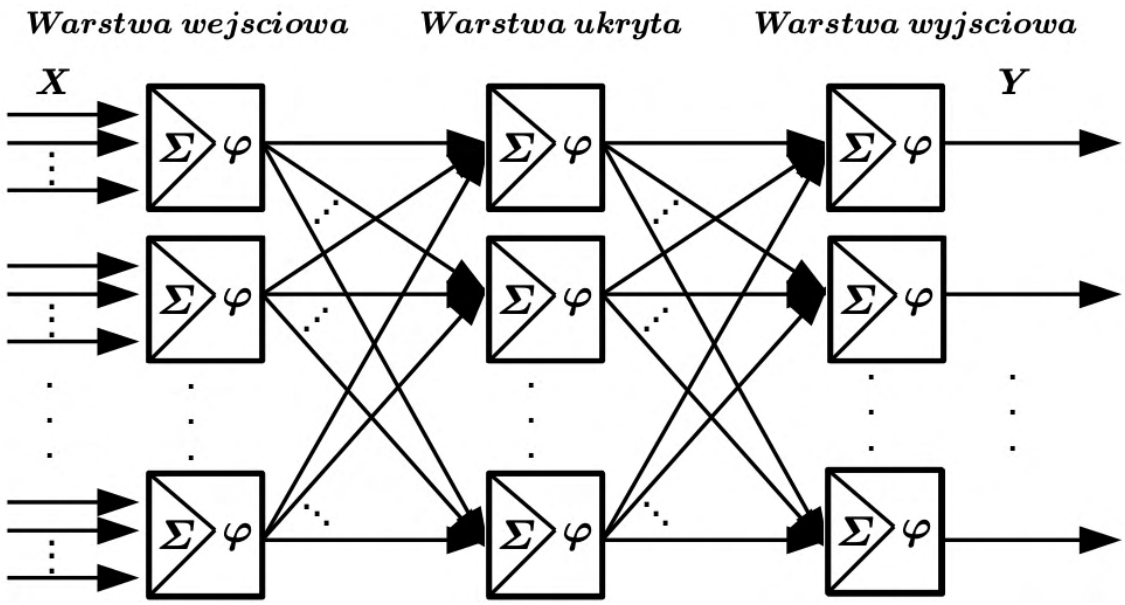
10 R. Tadeusiewicz, *Sieci Neuronowe*, Warszawa 1993, s. 47–48.
11 M. Minsky, S. Papert, *Perceptrons; an Introduction to Computational Geometry*, MIT Press 1969.



Rysunek 1.2.6. Schemat rozwiązań liniowych problemu funkcji OR i XOR.

Źródło: K.Y. Vestbøstad, *Spiking Neural Networks for Pattern Recognition*, The University of Bergen 2017.

Sieć nieliniowa. Przykładem sieci nieliniowej będzie poniższy schemat (rys. 1.2.7) modelu trójwarstwowego i jednokierunkowego (*feedforward*). Sieci wielowarstwowe i jednokierunkowe określane są na ogół jako wielowarstwowy perceptron (*multilayerd perceptron*, MLP).



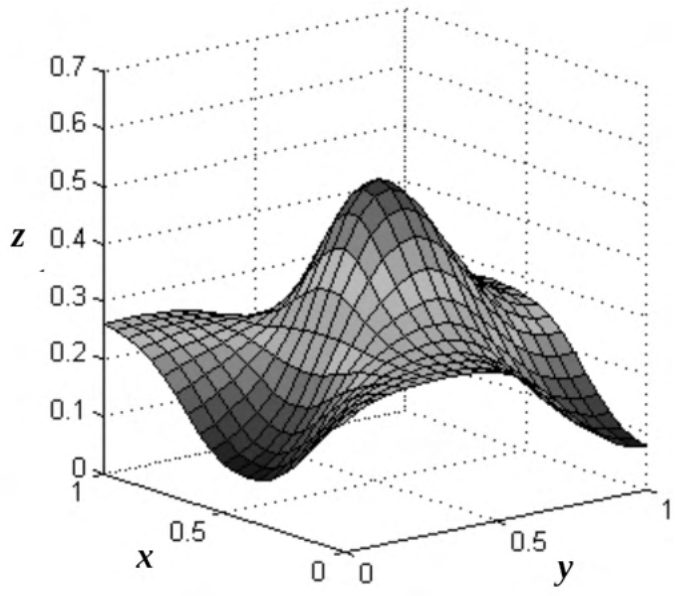
Rysunek 1.2.7. Schemat przykładowej sieci nieliniowej.

Źródło: Opracowanie własne.

Na rysunku 1.2.7, strzałki wewnątrz sieci tożsame są z połączeniami wagowymi, wielka sigma oznacza sumę iloczynów danych wejściowych i wag, zaś $\varphi()$ oznacza dowolną

funkcję aktywacji określającą sygnał wyjściowy neuronu. Określenie warstwa ukryta (*hidden layer)* jest uzasadnione faktem, iż sygnały wyjściowe tej warstwy podawane są wprost na warstwę kolejną, nie będąc przedmiotem obserwacji użytkownika.

Nieliniowa sieć wielowarstwowa dokonać może dowolnego powiązania sygnałów wejściowych z sygnałami wyjściowymi, tym samym wyodrębnienia dowolnych klas w przestrzeni klasyfikacji[12]. Jak wskazują Y. Hu i J. Hwang: „Zostało już udowodnione, że wyposażony w wystarczającą liczbę neuronów ukrytych perceptron wielowarstwowy, posiadający co najmniej dwie warstwy ukryte, zdolny jest do aproksymacji dowolnie złożonego odwzorowania obiektu w ramach skończonego procesu uczenia"[tłum. własne][13]. Innymi słowy, wielowarstwowy perceptron zdolny jest do dowolnego przekształcenia z przestrzeni danych wejściowych w przestrzeń sygnałów wyjściowych. Przykładem nieliniowego mapowania jest poniższa ilustracja (rys. 1.2.8.), gdzie dane wejściowe opisywane są na osiach x i y, zaś wyjściowe na osi z.



Rysunek 1.2.8. Uproszczony przykład mapowania obiektów w przestrzeni przez MLP.
Źródło: Y.H. Hu, J.-N. Hwang, *Handbook of neural network signal processing*, Boca Raton 2002, s. 18.

Jedna z podstawowych teorii opisujących sieci neuronowe dowodzi, że po spełnieniu odpowiednich warunków, nieliniowa sieć neuronowa jest uniwersalnym aproksymatorem (*universal approximator*)[14]. Teoria ta stanowi, że sieć zdolna jest aproksymować (łac. *approximare*, przybliżyć) dowolną funkcję, *i.e.* aproksymować

12 R. Tadeusiewicz, *Sieci Neuronowe*, Warszawa 1993, s. 52–54.
13 Y.H. Hu, J.-N. Hwang, *Handbook of neural network signal processing*, Boca Raton 2002, s. 17.
14 B.C. Csáji, *Approximation with Artificial Neural Networks*, Uniwersytet Eötvös Loránd 2001, s. 22.

odwzorowanie z dowolnego X do dowolnego Y. Ponieważ odwzorowanie $X \Rightarrow Y$ jest nieznane, to w procesie uczenia sieć dokonuje jego aproksymacji, modyfikując swoje parametry W, tak aby dla argumentów X, sygnały wyjściowe z sieci były możliwie przybliżone do oczekiwanych Y.

1.3. Uczenie sztucznej sieci neuronowej.

Uczenie nadzorowane. Najbardziej popularną metodą uczenia sztucznych sieci neuronowych jest klasyczny proces uczenia nadzorowanego, tzw. „uczenia z nauczycielem" (*supervised learning*), który omówiony zostanie na przykładzie sieci nieliniowych typu MLP. Ponieważ, stanowią one podstawowy typ sieci neuronowych, a przez prostą intuicję można odnieść ich przykład do innych architektur (podr. 1.4), gdyż równania będą sobie w naturalny sposób odpowiadać.

W ogólności, proces uczenia polega na wyznaczeniu oczekiwanego od sieci wektora sygnałów wyjściowych V, który ma być odpowiedzią na dany wektor sygnałów wejściowych X. Jeśli wektor sygnałów wyjściowych Y odbiega od oczekiwanego V, to dokonywana jest modyfikacja wag synaptycznych W metodą tzw. propagacji wstecznej (*backpropagation*), aby zbliżyć Y do V. Propagacja wsteczna jest konieczna w przypadku uczenia głębokich sieci neuronowych, aby określić jakiego sygnału oczekiwać na wyjściach neuronów ukrytych (*ergo* które spośród ich wag przyczyniły się do błędu i powinny ulec zmianie). Propagacja wsteczna posiada tę zaletę, że jest relatywnie szybka i odbywa się automatycznie. Zatem nadzór nauczyciela sprowadza się do opracowania zbioru danych uczących i zbioru przypisanych im właściwych odpowiedzi (nawet gdy nauczyciel nie wie dlaczego są właściwe). Na podstawie tych zbiorów, sieć w procesie uczenia powinna dokonać takiej generalizacji, aby udzielać odpowiedzi na dotąd nieznane X należące do jednej z wyuczonych klas. Nadmienić należy, że początkowa wartość wag synaptycznych przydzielana jest losowo w wybranym przedziale wartości.

Błąd określić można jedynie dla warstwy wyjściowej, bo nie jest możliwym ustalenie jakich sygnałów V należy oczekiwać na wyjściach warstw ukrytych. Więc obliczony na danym neuronie wyjściowym błąd δ, propagować należy wstecz dla ustalenia błędów poprzednich neuronów. Błąd danego neuronu wyjściowego przesyła

się wstecznie poprzez synapsy łączące go z poprzedzającymi neuronami, w celu obliczenia błędów tych neuronów, które to można dalej rzutować wstecz, aż do synaps na które sieć przyjmuje sygnał wejściowy.

Propagacja wsteczna nie będzie się więc różnic zasadniczo od zwyczajowego procesowania danych przez sieć. Dla uproszczenia, zbiory numerów neuronów określane będą ze względu na warstwę, jako o dla warstwy wyjściowej (*output*), h dla warstwy ukrytej (*hidden*) oraz i dla warstwy wejściowej (*input*). Dla ułatwienia przyjęto też, że $i = h = o = 1, 2, ..., n$. Rozpatrywana zatem będzie prosta sieć trójwarstwową, o równej liczbie neuronów wejściowych, ukrytych i wyjściowych. Celem uproszczenia, nie będzie na ogół oznaczany krok uczenia j, *i.e.* aktualnie rozważana przez sieć para wektorów $[X_j, V_j]^T$ ze względu na które wykonywana jest j-ta korekta wag synaptycznych (a która to para należy do danego zbioru uczącego $[[X_1, V_1]^T, \ldots, [X_k, V_k]^T]^T$, gdzie $j = 1, 2, ...k$). Nadmienić należy, że krok uczenia obejmować może więcej niż jedną parę wektorów $[X_j, V_j]^T$ na raz, co oznacza, że korekta wag synaptycznych odbywać się może ze względu na większą liczbę danych wejściowych jednocześnie, która to liczba określana jest jako *batch-size* (tutaj założono, że długość kroku, czyli *batch-size*, wynosi 1).

Dla aktualnie rozważanego neuronu warstwy wyjściowej, wyznaczyć można jego sygnał wyjściowy z zależności:

$$y_o = \varphi(\sum_{h=1}^{n} w_{ho} y_h) = \varphi(z_o) \qquad \text{(Równanie 1.3.1)}$$

Gdzie:

w_{ho} – wartość synapsy łączącej o-ty neuron warstwy wyjściowej z h-tym neuronem warstwy ukrytej;

y_h – sygnał wyjściowy z h-tego neurony warstwy ukrytej;

z_o – wartość pobudzenia o-tego neuronu warstwy wyjściowej;

y_o – sygnał wyjściowy z o-tego neuronu warstwy wyjściowej.

Na ogólny proces nauki nadzorowanej metodą propagacji wstecznej składają się zazwyczaj cztery następujące etapy[15]:

15 R. Rojas, *Neural Networks: A Systematic Introduction*, Springer Science & Business Media 2013, s. 166.

i) przednia propagacja danych przez kolejne warstwy sieci (oblicza się Y);

ii) obliczenie kosztu L na wyjściach sieci;

iii) propagacja wsteczna błędów neuronów δ przez kolejne warstwy sieci;

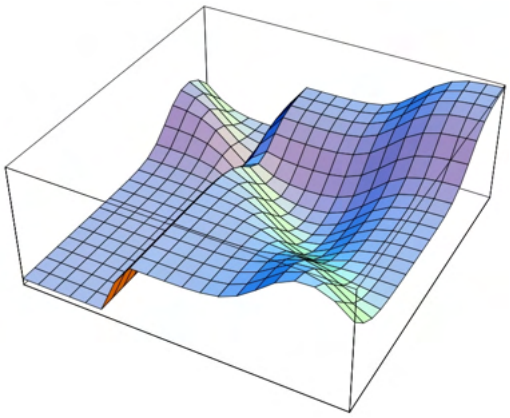
iv) korekta synaps Δw względem obliczonych błędów wszystkich neuronów.

Jak zauważa R. Rojas, istotą metody jest to, iż: „Algorytm propagacji wstecznej poszukuje minimum funkcji błędu w przestrzeni synaptycznej, metodą antygradientową. Kombinacja wartości synaps, która minimalizuje funkcję błędu, uznawana jest za rozwiązanie nauczanego problemu. Ponieważ rzeczona metoda wymaga obliczenia gradientu funkcji błędu dla każdego kroku uczenia, to zagwarantować musimy ciągłość i różniczkowalność funkcji błędu [*i.e.* przede wszystkim możliwość obliczenia pochodnej dla każdego argumentu funkcji]. [...] Różniczkowalność funkcji aktywacji sprawia, że funkcja obliczana przez sieć neuronową jest różniczkowalna (zakładając, że funkcja pobudzeń na każdym neuronie jest po prostu sumą ważonych sygnałów wejściowych), skoro sieć oblicza zaledwie kolejne funkcje złożone. Funkcja kosztu również staje się różniczkowalna. [...] W istocie, sieć reprezentuje łańcuch funkcji złożonych, który transformuje sygnał wejściowy na wektor wyjściowy (nazywany wzorcem). Sieć jest szczególną implementacją funkcji złożonej [która odwzorowuje sygnały] z przestrzeni wejściowej na wyjściową, którą nazywamy funkcją sieci ψ. Nauczany problem polega na odszukaniu optymalnej kombinacji wartości synaps, tak aby funkcja sieci ψ aproksymowała daną funkcję f tak blisko jak to możliwe. Jednakże nie znamy danej funkcji f wprost, a tylko pośrednio przez zbiór przykładów [uczących]”[tłum. własne][16]. Proces ten sprowadza się zatem do uzyskania dla sieci minimum określonej funkcji kosztu L (*loss function*)[17], zazwyczaj za pomocą metody antygradientowej. Ponieważ, gradient funkcji błędu (rys. 1.3.1) jest wskaźnikiem kierunku, w którym funkcja najszybciej rośnie (*gradient ascent*), a gradient ujemny (antygradient) wskaźnikiem kierunku, w którym najszybciej maleje (*gradient descent*)[18].

16 Ibid., s. 152–156.

17 K. Janocha, W.M. Czarnecki, *On Loss Functions for Deep Neural Networks in Classification*, arXiv, 18 lutego 2017 r., http://arxiv.org/abs/1702.05659.

18 D.E. Rumelhart, G.E. Hinton, R.J. Williams, *Learning representations by back-propagating errors*, „Nature” t. 323 nr 6088 (1986), DOI: 10.1038/323533a0.



Rysunek 1.3.1. Wizualizacja przykładowego gradientu funkcji błędu.

Źródło: R. Rojas, *Neural Networks: A Systematic Introduction*, Springer Science & Business Media 2013, s. 155.

Jedną z najpopularniejszych funkcji kosztu jest binarna entropia krzyżowa (*binary cross-entropy*), gdzie zakłada się istnienie tylko dwóch klas ($V = v = 0 \veebar 1$), rozstrzyganych przez jeden neurony wyjściowy. Koszt taki wynosi[19]:

$$L = -\left(v_o \ln(y_o) + (1 - v_o) \ln(1 - y_o)\right) \qquad \text{(Równanie 1.3.2)}$$

Gdzie:

L – koszt sieci (w danym kroku uczenia);

v_o – wartość oczekiwana od o-tego neuronu wyjściowego;

y_o – wartość otrzymana z o-tego neuronu wyjściowego;

$\ln()$ – logarytm naturalny.

Natomiast, gradient – zakładając tutaj numerację synaps niezależną od numeracji warstw i neuronów, gdzie $u = 1, 2, ..., p$ – przedstawić można następująco w notacji G. Leibniza[20]:

$$\nabla L = \left[\frac{\partial L}{\partial w_1}, \frac{\partial L}{\partial w_2}, \ldots, \frac{\partial L}{\partial w_p}\right]^T \qquad \text{(Równanie 1.3.3)}$$

Gradient powyższy stanowi wektor złożony z pochodnych cząstkowych ∂, który określa kierunek najszybszego wzrostu wartości funkcji błędu L ze względu na zmiany wartości synaps w_u. Generalnie rzecz ujmując, dla danej funkcji $f(x) = y$, pochodna $\frac{\partial y}{\partial x}$

19 P. Sadowski, *Notes on Backpropagation; University of California Irvine* [na:] https://www.ics.uci.edu/~pjsadows/notes.pdf, dostęp 28 listopada 2022 r.

20 S. Haykin, *Neural Networks: A Comprehensive Foundation*, Prentice Hall 1999, s. 172.

wskazuje jak szybko zmienia się wartość funkcji y ze względu na zmianę wartości jej argumentu x (zakładamy, że zmiana ta jest nieskończenie mała, tzw. różniczka)[21], gdzie ∂x oznacza zmianę wartości argumentu x, a ∂y oznacza zmianę wartości funkcji y, zaś pochodna $\frac{\partial y}{\partial x}$ wyznacza stosunek tych zmian. Ponieważ sieć neuronowa jest funkcją wielu zmiennych, a obliczać należy pochodne tylko względem wag synaptycznych w_u (bo tylko ich wartość poddawać należy korekcie), to stosuje się pochodne cząstkowe ∂ (w przeciwieństwie do pochodnej zupełnej d).

Reguła dla korekty wartości wag synaptycznych jest to tzw. reguła delta, której ogólna postać przedstawia się następująco[22]:

$$\Delta w_u = -\gamma \frac{\partial L}{\partial w_u} \qquad \text{(Równanie 1.3.4)}$$

gdzie γ (często też η lub α) jest arbitralne wyznaczonym współczynnikiem liczbowym określającym szybkość uczenia sieci (*learning rate*), *i.e.* współczynnikiem regulującym wielkość korekty w kierunku ujemnego gradientu (ujemnego ze względu na znak minus)[23]. Jeżeli w j-tym kroku uczenia wyliczono korektę Δw dla danej synapsy w, to ich suma da wartość tej synapsy w kolejnym kroku uczenia[24]:

$$w_{j+1} = w_j + \Delta w_j \qquad \text{(Równanie 1.3.5)}$$

Korekta dla warstwy wyjściowej. Zgodnie z regułą łańcuchową dla pochodnych funkcji złożonych (gdzie złożone są funkcje przetwarzania sygnałów przez neurony sieci), a po skróceniu dla neuronu wyjściowego[25]:

$$\frac{\partial L}{\partial w_{ho}} = \frac{\partial L}{\partial y_o} \frac{\partial y_o}{\partial z_o} \frac{\partial z_o}{\partial w_{ho}} \qquad \text{(Równanie 1.3.6)}$$

Gdzie:

$\partial L / \partial w_{ho}$ – jest pochodną funkcji kosztu L wobec synapsy w łączącej o-ty neuron wyjściowy z h-tym neuronem ukrytym.

21 R. Penrose, *Droga do Rzeczywistości*, Warszawa 2006, s. 101.
22 R. Rojas, *Neural Networks: A Systematic Introduction*, Springer Science & Business Media 2013, s. 157.
23 Ibid., s. 169.
24 R. Tadeusiewicz, *Sieci Neuronowe*, Warszawa 1993, s. 58.
25 Ibid.

$\partial L/\partial y_o$ – jest pochodną funkcji kosztu L wobec aktywacji o-tego neuronu y;

$\partial y_o/\partial z_o$ – jest pochodną aktywacji o-tego neuronu y wobec jego pobudzenia z;

$\partial z_o/\partial w_{ho}$ – jest pochodną pobudzenia o-tego neuronu z wobec synapsy w łączącej go z h-tym neuronem.

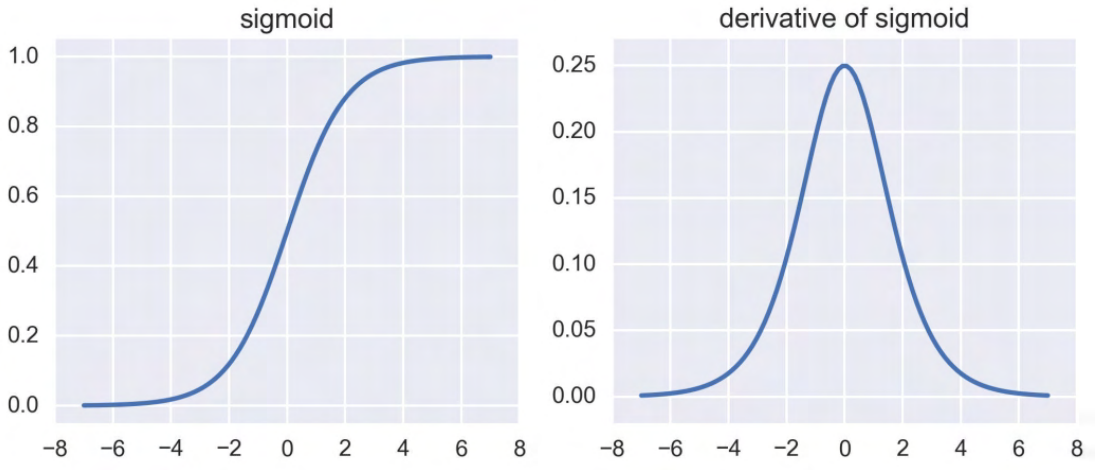
Innymi słowy, wartość funkcji kosztu L zależy od wartości sygnału wyjściowego y, który zależny jest od wartości pobudzenia neuronu z, który zależny jest z kolei od wartości danej wagi synaptycznej w.

Zakładając, że przykładowa sieć złożona jest z neuronów wyposażonych w sigmoidalną funkcję aktywacji, a funkcją kosztu jest binarna entropia krzyżowa, to pochodna funkcji koszu L wobec aktywacji o-tego neuronu wyjściowego y wyniesie[26]:

$$\frac{\partial L}{\partial y_o} = \frac{y_o - v_o}{y_o(1 - y_o)} \qquad \text{(Równanie 1.3.7)}$$

Natomiast pochodna aktywacji y danej funkcją sigmoidalną (rys. 1.3.2) względem pobudzenia o-tego neuronu z przybierze postać[27]:

$$\frac{\partial y_o}{\partial z_o} = \frac{\partial \varphi(z_o)}{\partial z_o} = y_o(1-y_o) \qquad \text{(Równanie 1.3.8)}$$



Rysunek 1.3.2. Wykres funkcji sigmoidalnej (*sigmoid*) i jej pochodnej (*derivative of sigmoid*). Gdzie wartość pochodnej rośnie, kiedy funkcja rośnie szybciej, zaś maleje, kiedy funkcja rośnie wolniej.

Źródło: A. Glassner, *Deep Learning: From Basics to Practice*, t. 1, Seattle 2018, s. 769.

26 P. Sadowski, *Notes on Backpropagation; University of California Irvine* [na:] https://www.ics.uci.edu/~pjsadows/notes.pdf, dostęp 28 listopada 2022 r.

27 C.M. Bishop, *Neural Networks for Pattern Recognition*, Clarendon Press 1995, s. 145.

Ostatecznie, ze względu na prostą proporcjonalność neuronu liniowego[28]:

$$\frac{\partial z_o}{\partial w_{ho}} = y_h \qquad \text{(Równanie 1.3.9)}$$

Rozpoczynając propagację wsteczną, ustalić należy dla każdego neuronu jego δ gradient lokalny błędu[29] (często określany – także w pracy niniejszej – jako błąd neuronu). Dla neuronów wyjściowych wymaga to uwzględnienia pochodnej kosztu względem aktywacji oraz pochodnej aktywacji względem pobudzenia danego neuronu. Rzutuje się zatem koszt odpowiedzi na neuron, który jej udzielił, uwzględniając przy tym jego nieliniowość. Stąd uzyskuje się δ_o konieczny dla późniejszej korekty synaps w_{ho}, łączących rozważany neuron z neuronami warstwy poprzedniej[30]:

$$\begin{aligned} \frac{\partial L}{\partial y_o}\frac{\partial y_o}{\partial z_o} &= \frac{y_o - v_o}{y_o(1-y_o)}\ (y_o(1-y_o)) \\ &= y_o - v_o \\ &= \delta_o \end{aligned} \qquad \text{(Równanie 1.3.10)}$$

Stąd pochodne cząstkowe gradientu zredefiniować można następująco[31]:

$$\frac{\partial L}{\partial w_{ho}} = \delta_o y_h \qquad \text{(Równanie 1.3.11)}$$

Reguła delta dla warstwy wyjściowej przybierze postać[32]:

$$\Delta w_{ho} = -\gamma \delta_o y_h \qquad \text{(Równanie 1.3.12)}$$

Korekta dla warstw ukrytych. Aby dokonać aktualizacji połączeń synaptycznych między warstwą ukrytą a wejściową w_{ih}, rozłożyć należy pochodne funkcji błędu L względem tych połączeń, które (po skróceniu dla neuronu ukrytego)[33]:

28 R. Tadeusiewicz, *Sieci Neuronowe*, Warszawa 1993, s. 58–60.
29 S. Haykin, *Neural Networks: A Comprehensive Foundation*, Prentice Hall 1999, s. 185.
30 P. Sadowski, *Notes on Backpropagation; University of California Irvine* [na:] https://www.ics.uci.edu/~pjsadows/notes.pdf, dostęp 28 listopada 2022 r.
31 R. Rojas, *Neural Networks: A Systematic Introduction*, Springer Science & Business Media 2013, s. 164.
32 Ibid.
33 C.M. Bishop, *Neural Networks for Pattern Recognition*, Clarendon Press 1995, s. 143; S. Haykin, *Neural Networks: A Comprehensive Foundation*, Prentice Hall 1999, s. 188.

$$\frac{\partial L}{\partial w_{ih}} = \frac{\partial L}{\partial y_h}\frac{\partial y_h}{\partial z_h}\frac{\partial z_h}{\partial w_{ih}} \qquad \text{(Równanie 1.3.13)}$$

Ponieważ nie wiadomo jakich sygnałów wyjściowych oczekiwać od neuronów ukrytych, to chcąc uzyskać błąd lokalny δ_h danego neuronu ukrytego, należy rzutować nań wszystkie błędy δ_o z warstwy ukrytej, określając stąd w jakim stopniu się do nich przyczynił. Po zastosowaniu reguły łańcuchowej, pochodna funkcji kosztu L wobec aktywacji h-tego neuronu ukrytego y wyniesie[34]:

$$\begin{aligned}\frac{\partial L}{\partial y_h} &= \sum_{o=1}^{n}\frac{\partial L}{\partial y_o}\frac{\partial y_o}{\partial z_o}\frac{\partial z_o}{\partial y_h} \\ &= \sum_{o=1}^{n}\delta_o\frac{\partial z_o}{\partial y_h} \\ &= \sum_{o=1}^{n} w_{ho}\delta_o\end{aligned} \qquad \text{(Równanie 1.3.14)}$$

Rozwiązanie dla pierwszych dwóch pochodnych znane jest z warstwy wyjściowej, zaś rozwiązanie pochodnej $\partial z_o/\partial y_h$ wynika z proporcjonalności neuronu liniowego. Jak można zaobserwować, równanie powyższe jest podobne do definicji neuronu, aby to dodatkowo podkreślić, w_{ho} i δ_o zostały w ostatnim wierszu zamienione miejscami.

Pochodna aktywacji y wobec pobudzenia h-tego neuronu ukrytego z (zakładając ponownie sigmoidalną funkcję aktywacji), będzie podobna jak dla warstwy wyjściowej[35]:

$$\frac{\partial y_h}{\partial z_h} = \frac{\partial \varphi(z_h)}{\partial z_h} = y_h(1-y_h) \qquad \text{(Równanie 1.3.15)}$$

Stąd zdefiniować można błąd lokalny δ_h jako[36]:

$$\begin{aligned}\frac{\partial L}{\partial y_h}\frac{\partial y_h}{\partial z_h} &= \left(\sum_{o=1}^{n} w_{ho}\delta_o\right) y_h(1-y_h) \\ &= \delta_h\end{aligned} \qquad \text{(Równanie 1.3.16)}$$

34 C.M. Bishop, *Neural Networks for Pattern Recognition*, Clarendon Press 1995, s. 143; S. Haykin, *Neural Networks: A Comprehensive Foundation*, Prentice Hall 1999, s. 188.
35 C.M. Bishop, *Neural Networks for Pattern Recognition*, Clarendon Press 1995, s. 145.
36 Ibid., s. 146.

Ostatecznie, pochodna pobudzenia h-tego neuronu ukrytego z względem synapsy w łączącej go z i-tym neuronem wejściowym ma postać[37]:

$$\frac{\partial z_h}{\partial w_{ih}} = y_i \qquad \text{(Równanie 1.3.17)}$$

Stąd pochodną funkcji kosztu L dla synapsy w łączącej i-ty neuron wejściowy z h-tym neuronem ukrytym zredefiniować można następująco[38]:

$$\frac{\partial L}{\partial w_{ih}} = \delta_h y_i \qquad \text{(Równanie 1.3.18)}$$

Reguła delta dla warstwy ukrytej przybierze postać[39]:

$$\Delta w_{ih} = -\gamma \delta_h y_i \qquad \text{(Równanie 1.3.19)}$$

Jak podsumował S. Haykin: „Po pierwsze, korekta $\Delta w_{ji}(n)$ synapsy łączącej i-ty neuron z j-tym neuronem jest zdefiniowana regułą delta:

$$\begin{pmatrix}\text{korekta wagi}\\ \text{synaptycznej}\\ \Delta w_{ji}(n)\end{pmatrix} = \begin{pmatrix}\text{współczynnik}\\ \text{uczenia}\\ -\eta\end{pmatrix}\begin{pmatrix}\text{gradient}\\ \text{lokalny}\\ \delta_j(n)\end{pmatrix}\begin{pmatrix}\text{sygnał wejściowy}\\ \text{do j-tego neuronu}\\ y_i(n)\end{pmatrix}$$

Po drugie, gradient lokalny $\delta_j(n)$ zależy od tego, czy j-ty neuron jest wyjściowy czy ukryty:

> i) Jeżeli j-ty neuron jest wyjściowy, to $\delta_j(n)$ jest równa iloczynowi pochodnej jego aktywacji i pochodnej kosztu, które związane są z j-tym neuronem.
>
> ii) Jeżeli j-ty neuron jest ukryty, to $\delta_j(n)$ jest równa iloczynowi pochodnej jego aktywacji i ważonej sumy [gradientów lokalnych] δ, które obliczone zostały na neuronach następującej warstwy ukrytej lub warstwy wyjściowej, połączonych z j-tym neuronem [tłum. własne]”[40].

37 R. Tadeusiewicz, *Sieci Neuronowe*, Warszawa 1993, s. 58–60.
38 C.M. Bishop, *Neural Networks for Pattern Recognition*, Clarendon Press 1995, s. 146.
39 R. Rojas, *Neural Networks: A Systematic Introduction*, Springer Science & Business Media 2013, s. 169.
40 S. Haykin, *Neural Networks: A Comprehensive Foundation*, Prentice Hall 1999, s. 188–189.

Korekty połączeń synaptycznych wykonuje się dopiero po przeprowadzeniu kompletnej: i) propagacji przedniej; ii) obliczeniu kosztu; iii) propagacji wstecznej. Jak zauważa R. Rojas: „Bardzo istotnym jest, aby dokonać korekty wag tylko i wyłącznie po obliczeniu rzutowanych wstecznie błędów dla wszystkich parametrów sieci. W innym przypadku, korekty stają się uwikłane w propagację wsteczną błędu i wyliczane korekty nie korespondują już z negatywnym kierunkiem gradientu [tłum. własne][41]".

Uczenie nienadzorowane. Metoda uczenia nienadzorowanego (*unsupervised learning*), tzw. uczenie bez nauczyciela, znajduje najczęstsze zastosowanie w sytuacjach, gdy nie wiadomo, jakie są prawidłowe odpowiedzi lub ustalenie ich byłoby zbyt kosztowne. Posłużono się poniżej przykładem klasycznej metody uczenia nienadzorowanego dla sieci liniowych, *i.e*. metodą Hebba (*Hebbian learning*)[42]. Jest ona przeniesieniem na grunt sztucznych sieci neuronowych teorii D. Hebba, dotyczącej procesów uczenia się biologicznych sieci neuronowych (*Hebbian theory*). Teoria ta często ujmowana jest powiedzeniem, że: „Neurony łączą się razem, jeżeli są wspólnie aktywne [*neurons that fire together, wire together*; tłum. własne]"[43]. Innymi słowy, jeżeli neurony, które połączone są razem synapsą, reagują wspólnie na dany sygnał, to każda taka reakcja wzmocni ich połączenie synaptyczne[44]. Przede wszystkim, nie ma tutaj żadnego wzorca właściwych odpowiedzi V, zamiast tego stosuje się algorytm uczenia, który wzmacniał będzie połączenia na synapsach łączących te neurony, które okazały się symultanicznie podatne na dany X, a w konsekwencji na inne X należące do tej samej klasy. Zakłada się zatem, że ze względu na odmienność klas i podobieństwo X wewnątrz klas, wzmacnianie wspólnych reakcji neuronów doprowadzi do ekskluzywnego wyczulenia ich na daną klasę.

Równanie neuronu liniowego przedstawić można następująco, oznaczając numery neuronów jako $u = 1, 2, \ldots, p$, numery kroków uczenia jako $j = 1, 2, \ldots, k$, zaś numery elementów wektora wejściowego $X^{(j)}$ jako $i = 1, 2, \ldots, n$:

41 R. Rojas, *Neural Networks: A Systematic Introduction*, Springer Science & Business Media 2013, s. 169.

42 B.A. Olshausen, *Linear Hebbian learning and PCA; Redwood Center for Theoretical Neuroscience at the University of California in Berkeley* [na:] https://redwood.berkeley.edu/wp-content/uploads/2018/08/handout-hebb-PCA.pdf, 2012 r., dostęp 3 listopada 2022 r.

43 S. Löwel, W. Singer, *Selection of Intrinsic Horizontal Connections in the Visual Cortex by Correlated Neuronal Activity*, „Science" t. 255 nr 5041 (1992), DOI: 10.1126/science.1372754, s. 211.

44 D. Hebb, *The Organization of Behavior: A Neuropsychological Theory*, New York 1949, s. 62.

$$y^{(u)(j)} = \sum_{i=1}^{n} w_i^{(u)(j)} x_i^{(j)} \qquad \text{(Równanie 1.3.20)}$$

Gdzie:

$y^{(u)(j)}$ – sygnał wyjściowy u-tego neuronu w j-tym kroku uczenia;

$w_i^{(u)(j)}$ – i-ta waga synaptyczna u-tego neuronu w j-tym kroku uczenia;

$x_i^{(j)}$ – i-ty element wektora wejściowego $X^{(j)}$ w j-tym kroku uczenia.

Reguła Hebba dla aktualizacji wag synaptycznych przybierze następującą postać[45]:

$$w_i^{(u)(j+1)} = w_i^{(u)(j)} + \gamma \; x_i^{(j)} y^{(u)(j)} \qquad \text{(Równanie 1.3.21)}$$

Wedle której, wartość synapsy $w_i^{(u)}$ znajdującej się na i-tym wejściu u-tego neuronu, wzrasta po prezentacji j-tego wektora wejściowego $X^{(j)}$, proporcjonalnie do iloczynu jego i-tej składowej $x_i^{(j)}$ (która przechodzi rozważaną synapsą na rozważany neuron) i sygnału wyjściowego $y^{(u)(j)}$ z rozważanego u-tego neuronu. Gdzie γ ponownie oznacza współczynnik szybkości uczenia. Jak streścił to B. Olshausen: „Reguła Hebba stanowi, że waga synaptyczna łącząca dwa neurony powinna wzrastać proporcjonalnie do korelacji pomiędzy pre-synaptyczną i post-synaptyczną aktywnością. Zatem, dla neuronu liniowego, każda waga w_i powinna wzrosnąć proporcjonalnie do korelacji pomiędzy y oraz x_i[tłum. własne]”[46].

Tak więc, metoda działania sieci nienadzorowanej polega na grupowaniu (*clustering*) podobnych sygnałów wejściowych poprzez wzmacnianie neuronów, które najbardziej reagują na ich obecność. Jak zauważa R. Tadeusiewicz: „[Rezultat nienadzorowanego uczenia] dość istotnie zależy od początkowego stanu sieci (początkowych, przypadkowych wartości $w_i^{(u)(1)}$) decydującego o tym, jak w początkowym etapie uczenia zaczną krystalizować się ośrodki przyszłych grup. Niewielka jest szansa na to, by różne klasy podobieństwa istniejące wśród wejściowych sygnałów od razu odnalazły „swoje” neurony, zatem trzeba się liczyć z tym, że na tę

45 R. Tadeusiewicz, *Sieci Neuronowe*, Warszawa 1993, s. 37.

46 B.A. Olshausen, *Linear Hebbian learning and PCA; Redwood Center for Theoretical Neuroscience at the University of California in Berkeley* [na:] https://redwood.berkeley.edu/wp-content/uploads/2018/08/handout-hebb-PCA.pdf, 2012 r., dostęp 3 listopada 2022 r.

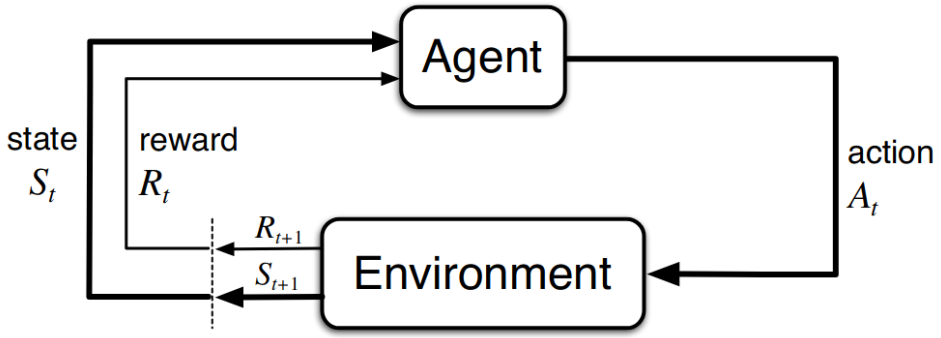
samą klasę „wskazywać" będzie kilka neuronów. Oznacza to, że liczba neuronów w sieci p musi być większa niż oczekiwana liczba rozróżnialnych klas, a ponadto nie pozwala z góry przewidzieć, który neuron nauczy się sygnalizować którą klasę"[47].

W porównaniu do metody nadzorowanej, ucząc metodą nienadzorowaną: i) dysponować należy większą ilością danych; ii) sieć neuronowa musi być większych rozmiarów; ii) uczenie sieci trwa dłużej; iii) oraz większe jest ryzyko niepowodzenia, skutkujące wielokrotnym ponawianiem uczenia. Przewagą sieci nienadzorowanych jest jednak to, że dane nie muszą być opisane, *i.e.* nie musi się znać właściwych odpowiedzi. Jak argumentuje R. Strom: „Postulat Hebba, a podstawa Hebbowskiej metody uczenia, sugeruje biologicznie prawdopodobną metodę aktualizacji wag synaptycznych dla sieci neuronowych. Co więcej reguła Hebba polega na samo-organizacji i lokalności praw [wewnątrz sieci], sugerując, że udany algorytm Hebbowski ma większe szanse na stworzenie inteligentnego systemu, niż algorytm nadzorowanego uczenia, jeżeli tylko zaakceptujemy hipotezę, że inteligencja jest emergentną cechą systemów złożonych [tłum. własne]"[48].

Uczenie przez wzmacnianie. Jako trzecią metodę uczenia sieci neuronowych wymienia się uczenie przez wzmacnianie (*reinforcement learning*), które również opiera się na metodzie antygradientowej (gdy dotyczy sieci neuronowych)[49]. W metodzie tej (rys. 1.3.3): i) sieć neuronowa pełni rolę agenta (*agent*); ii) który operując w danym środowisku (*environment*); iii) podejmuje interakcje z tym środowiskiem (*actions*); iv) przez co zmienia jego stan (*state*); v) uzyskując stąd nagrodę (*reward*); vi) na podstawie której dokonuje aktualizacji swojej polityki (*policy*), czyli wyuczonych odwzorowań z poznanych stanów na dozwolone akcje, aby wybrać taką akcję w kolejnym kroku, która pozwoli zmaksymalizować jego nagrodę. Na przykładzie słynnej sieci neuronowej *AlphaZero*[50]: i) środowiskiem jest gra w szachy; ii) agentem jest sieć neuronowa; iii) akcjami są dozwolone ruchy; iv) stanem jest układ figur na szachownicy (zmieniający się w reakcji na akcje agenta); v) polityką jest wektor wyjściowy z sieci,

47 R. Tadeusiewicz, *Sieci Neuronowe*, Warszawa 1993, s. 38.
48 R.W. Strom, *Hebbian Learning in Multilayer Neural Networks* [w:] Los Angeles 2007, s. ii.
49 R.S. Sutton, A.G. Barto, *Reinforcement Learning: An Introduction*, Cambridge 2015.
50 D. Silver, T. Hubert, J. Schrittwieser, I. Antonoglou, M. Lai, A. Guez, M. Lanctot, L. Sifre, D. Kumaran, T. Graepel, T. Lillicrap, K. Simonyan, D. Hassabis, *Mastering Chess and Shogi by Self-Play with a General Reinforcement Learning Algorithm*, „arXiv:1712.01815 [cs]" (2017), http://arxiv.org/abs/1712.01815.

przypisujący prawdopodobieństwa do akcji dozwolonych w danym stanie; vi) wartością (*value*) jest przewidywany przez sieć rezultat rozgrywki z punktu widzenia stanu w którym się znajduje; vii) nagrodę wylicza funkcja kosztu penalizująca: vii.i) różnicę pomiędzy rzeczywistym rezultatem rozgrywki, a wartością oszacowaną przez sieć; vii.ii) różnicę pomiędzy polityką a rozkładem prawdopodobieństwa nad potencjalnymi sekwencjami akcjami i reakcji, uzyskanym metodą *Monte-Carlo Tree Search*.



Rysunek 1.3.3. Schemat uczenia przez wzmacnianie, gdzie: *agent* to agent, *environment* to środowisko, *action* to akcja, *state* to stan, *reward* to nagroda, krok oznaczony jest jako t.

Źródło: R.S. Sutton, A.G. Barto, *Reinforcement Learning: An Introduction*, Cambridge 2015, s. 54.

Uszczegóławiając przykład sieci *AlphaZero*, definiowana jest jako:

$$f_\theta(s) = (P, v) \qquad \text{(Równanie 1.3.22)}$$

Gdzie:

f – sieć neuronowa;

θ – aktualne wagi synaptyczne sieci;

s – obserwowany stan środowiska;

v – wartość, czyli wynik rozgrywki przewidywany przez sieć w danym stanie s;

P – polityka, czyli wektor, którego elementami są prawdopodobieństwa p sukcesu dozwolonych akcji a w obserwowanym stanie s, gdzie $a = 1, 2, \ldots, m$, zaś $p_a = Pr(a \mid s)$ (oszacowane przez sieć neuronową).

Natomiast funkcja kosztu, na podstawie której dokonuje się gradientowej aktualizacji wag synaptycznych sieci, przedstawia się następująco:

$$L = (z - v)^2 - \pi^T \ln(P) + c \|\theta\|^2 \qquad \text{(Równanie 1.3.23)}$$

Gdzie:

L – funkcja kosztu;

z – rzeczywisty wynik rozgrywki (wygrana to 1, remis to 0, przegrana to -1);

v – wartość, wynik rozgrywki oszacowany przez sieć w przedziale $[-1, 1]$;

P – polityka;

π – rozkład prawdopodobieństwa nad potencjalnymi sekwencjami akcjami i reakcji, uzyskany metodą *Monte-Carlo Tree Search*;

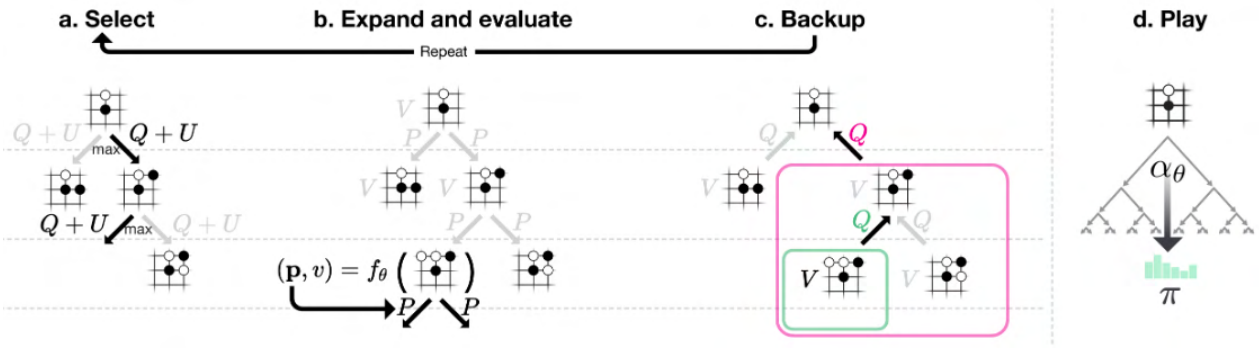
θ – aktualne wagi synaptyczne sieci;

$\|\theta\|^2$ – regularyzacja wag metodą L_2, która jest sumą ich kwadratów;

c – parametr regulujący stopień regularyzacji połączeń synaptycznych.

Ponieważ ilość możliwych rozgrywek szachowych wynosi co najmniej 10^{120} (tzw. liczba Shannona), to ustalenie kompletnego drzewa gry (*game tree*), które by je wszystkie opisywało, byłoby niepraktyczne. Zamiast tego stosuje się metodę *Monte-Carlo Tree Search* (rys. 1.3.4), gdzie w uproszczeniu: i) określa się, jakie są możliwe akcje w danym stanie s (korzeń drzewa; *root*); ii) wybiera się taką akcję a, dla której sieć neuronowa $f_\theta(s)$ przewiduje najwyższe prawdopodobieństwo p_a i wartość v; iii) w wyniku tej akcji dociera się do nowego stanu, z którego dalej rozgałęzia się drzewo (rozwinięcie; *expansion*), powtarzając punkt pierwszy i drugi, aż dotrze się do stanu, za którym istnieją nieodwiedzone wcześniej stany (liść drzewa; *leaf*); iv) aktualizuje się prawdopodobieństwa na podstawie (P, v) obliczonych przez sieć dla liścia drzewa, uzyskując ich nowy rozkład π (propagacja wsteczna; *backup*)[51]. Przed wykonaniem akcji przeprowadza się 1600 równoległych symulacji metodą *Monte-Carlo Tree Search*, gdzie *AlphaZero* sama jest swoim przeciwnikiem, a podjęta zostaje ta akcja, która była najczęściej rozwijana podczas symulacji. Celem uczenia jest zbliżenie (P, v) do (π, z), gdzie *Monte-Carlo Tree Search* jest operatorem poprawiającym politykę (*policy improvement operator*).

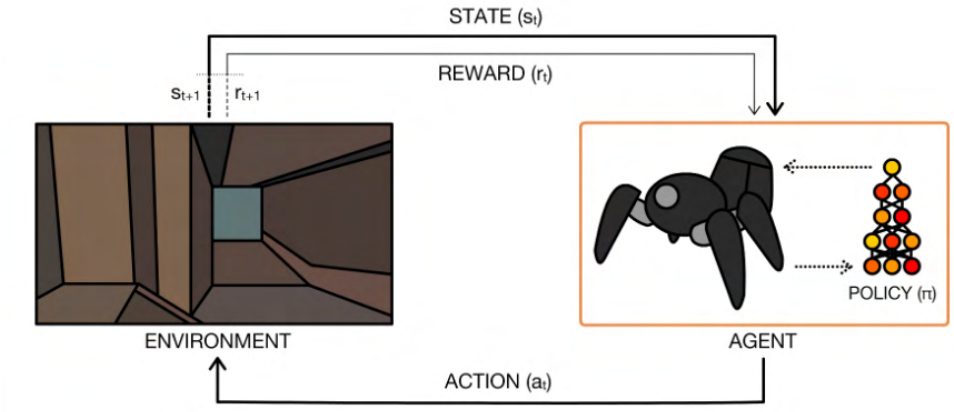
51 D. Silver, J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang, A. Guez, T. Hubert, L. Baker, M. Lai, A. Bolton, Y. Chen, T. Lillicrap, F. Hui, L. Sifre, G. van den Driessche, T. Graepel, D. Hassabis, *Mastering the game of Go without human knowledge*, „Nature" t. 550 nr 7676 (2017), DOI: 10.1038/nature24270.



Rysunek 1.3.4. Przykładowy schemat *Monte-Carlo Tree Search* dla *AlphaZero*, gdzie: i) *select* (wybór) to etap, na którym sieć wybiera akcję z danego stanu, w którym się znajduje; ii) *expand* and *evaluate* (rozwinięcie i ewaluacja), to etap, na którym drzewo rozwijane jest do liścia, którego stan ewaluowany jest przez sieć pod kątem kolejnej akcji; iii) *backup* (propagacja wsteczna), to etap na którym rozkład prawdopodobieństwa nad akcjami jest wstecznie aktualizowany, aby uzyskać π; iv) *play* (rozgrywka), to etap na którym podejmowana ta jest akcja, która była najczęściej rozwijana.

Źródło: D. Silver, J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang, A. Guez, T. Hubert, L. Baker, M. Lai, A. Bolton, Y. Chen, T. Lillicrap, F. Hui, L. Sifre, G. van den Driessche, T. Graepel, D. Hassabis, Mastering the game of Go without human knowledge, „Nature" t. 550 nr 7676 (2017), DOI: 10.1038/nature24270.

Modele uczone przez wzmacnianie znajdują zastosowania między innymi w grach planszowych i wideo, robotyce (rys. 1.3.5), przetwarzaniu języka naturalnego, oraz przetwarzaniu obrazów i materiałów wideo[52].



Rysunek 1.3.5. Przykładowy schemat uczenia przez wzmacnianie, gdzie *policy* to polityka, *agent* to agent, *action* to akcja, *reward* to nagroda, *state* to stan, zaś *environment* to środowisko.

Źródło: K. Arulkumaran, M.P. Deisenroth, M. Brundage, A.A. Bharath, *A Brief Survey of Deep Reinforcement Learning*, „IEEE Signal Processing Magazine" t. 34 nr 6 (2017), DOI: 10.1109/MSP.2017.2743240.

52 Y. Li, *Deep Reinforcement Learning: An Overview*, arXiv, 25 listopada 2018 r., http://arxiv.org/abs/1701.07274.

Jak zauważają R. Sutton i A. Barto: „Uczenie przez wzmacnianie jest różne od uczenia nadzorowanego [...] które stanowi istotną metodę uczenia, ale nieadekwatną w przypadku uczenia poprzez interakcje. W przypadku problemów interaktywnych, pozyskanie przykładów oczekiwanych akcji, które są zarazem prawidłowe i reprezentatywne dla wszystkich stanów w których agent może się znaleźć, jest często niepraktyczne. [...] Uczenie przez wzmacnianie jest także różne od [...] uczenia nienadzorowanego. [...] Choć można się skusić o stwierdzenie, że uczenie przez wzmacnianie jest rodzajem uczenia nienadzorowanego, ponieważ nie polega na przykładach prawidłowych akcji, to uczenie przez wzmacnianie dąży do maksymalizacji nagrody, raczej niż odnalezienia ukrytej struktury [danych]. Tak więc uważamy uczenie przez wzmacnianie za trzecią metodą uczenia maszynowego, obok uczenia nadzorowanego i nienadzorowanego [...]”[tłum. własne][53].

Podstawowym wyzwaniem uczenia przez wzmacnianie jest zdefiniowanie skutecznego systemu nagród, a w dalszej kolejności osiągnięcie równowagi pomiędzy eksploracyjnymi i eksploatacyjnymi zachowaniami agenta. Zachowania eksploatacyjne (*exploitation*) polegają na wykonywaniu akcji, które dotychczas przynosiły nagrodę. Natomiast zachowania eksploracyjne (*exploration*), polegają na poszukiwaniu nowych akcji, które być może przynosić będą większą nagrodę. Tak więc, aby pozyskiwać nagrody agent musi stosować najlepsze znane mu akcje, ale by poznać te akcje, musi się on podjąć eksploracji, testując nowe akcje kosztem aktualnych nagród[54].

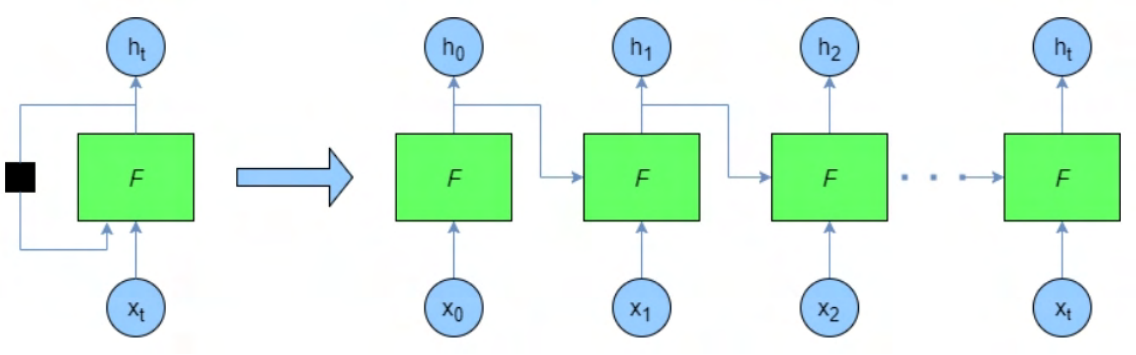
1.4. Architektury sztucznych sieci neuronowych.

Sieci perceptronowe. Liniowe i nieliniowe sieci, które są: i) jednokierunkowe, *i.e*. sygnały przesyłane są kolejnymi warstwami, od wejściowej do wyjściowej; ii) każdy neuron wejściowy otrzymuje na swoje synapsy całość danych wejściowych (ma tyle synaps co ma elementów wektor wejściowy); iii) każdy neuron danej warstwy jest połączony swoim wejściem z wyjściami wszystkich neuronów warstwy poprzedzającej (ma tyle synaps co poprzednia warstwa neuronów); iv) oraz połączony jest swoim wyjściem z wejściami wszystkich neuronów warstwy następującej. Perceptron był pierwszą w historii siecią neuronową, a najpopularniejszym

53 R.S. Sutton, A.G. Barto, *Reinforcement Learning: An Introduction*, Cambridge 2015, s. 16–17.
54 Ibid.

modelem tego typu jest nieliniowy perceptron wielowarstwowy (*multilayerd perceptron*, MLP). Na przykładzie sieci perceptronowych omówiono zagadnienia budowy i uczenia sztucznych sieci neuronowych w podrozdziałach poprzedzających (podr. 1.2 i 1.3).

Sieci rekurencyjne. Jednym z pierwszych kroków w ewolucji sieci perceptronowych w inne architektury, były sieci rekrurencyjne *(recurrent neural networks*), które tym różnią się od sieci jednokierunkowych (*feedforward*), iż pozwalają na wsteczne przesyłanie sygnałów przez neurony (*feedback*). Jak zauważają R. Pascanau, Y. Bengio i T. Milkova: „Struktura sieci [rekurencyjnej] jest bardzo podoba do standardowej struktury wielowarstwowego perceptronu, z tą różnicą, że dopuszczamy między neuronami ukrytymi połączenia tożsame z opóźnieniem czasowym [poprzez umożliwianie tym neuronom odsyłania swoich sygnałów wyjściowych na wejścia neuronów warstw poprzedzających]. Poprzez te połączenia sieć może zachowywać informacje o przeszłych sygnałach wejściowych, co pozwala jej na odkrycie korelacji czasowych pomiędzy zdarzeniami, które w zbiorze danych uczących są od siebie najpewniej znacznie oddalone (jest to właściwość krytyczna dla skutecznej nauki szeregów czasowych)[tłum. własne]”[55].

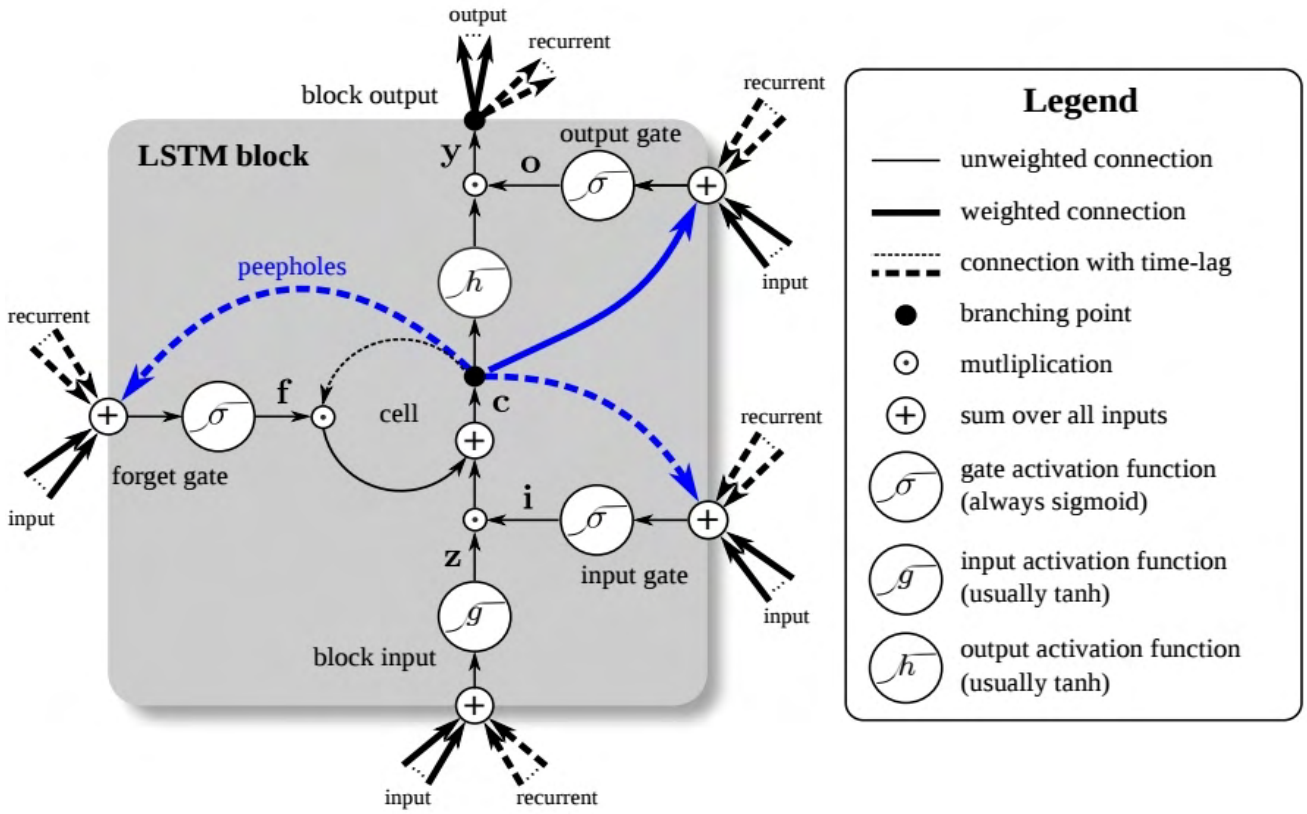


Rysunek 1.4.1. Schemat symboliczny rozwinięcia sieci rekurencyjnej F, gdzie sygnał wyjściowy h_i z sieci, przesyłany jest t-krotnie na jej wejście, wraz z kolejnymi sygnałami wejściowymi x_i, aż do uzyskania sygnału wyjściowego h_t ($i = 1, 2, ..., t$).

Źródło: E. Dogariu, S. Garg, B. Khadan, A. Potts, M. Scornavacca, *Using Machine Learning to Correlate Twitter Data and Weather Patterns* [w:] *2019 IEEE MIT Undergraduate Research Technology Conference (URTC)*, 2019.

55 R. Pascanu, T. Mikolov, Y. Bengio, *On the difficulty of training Recurrent Neural Networks*, arXiv, 15 lutego 2013 r., http://arxiv.org/abs/1211.5063, s. 1.

Klasycznym, najbardziej generalnym przykładem sieci rekurencyjnej jest wariant w pełni połączony (*fully recurrent neural network*)[56], gdzie połączone są ze sobą wszystkie neurony sieci, *i.e.* wyjście każdego neuronu dociera na wejście każdego innego neuronu, niezależnie od jego warstwy, oraz na swoje własne wejście. Najpopularniejszym zaś wariantem sieci rekurencyjnych są modele składające się z jednostek (*units*) *long short-term memory* (LSTM), które w 1997 roku opracowali1 S. Hochreiter i J. Schmidhuber[57]. Powodem dla opracowania sieci typu LSTM był problem zanikającego gradientu, którego wartość wykładniczo maleje, aż $\Delta w \approx 0$ (w przeciwieństwie do eksplodującego gradientu [*exploding gradient*], który również uniemożliwia naukę modelu, przyrastając wykładniczo)[58].



Rysunek 1.4.2. Schemat jednostki LSTM, gdzie: i) *unweighted connection* to połączenie nieważone; ii) *weighted connection* to połączenie ważone; iii) *connection with time-lag* to połączenie z opóźnieniem, *i.e.* rekursywne; iv) *branching point* to punkt rozgałęzienia połączeń; v) *multiplication* to iloczyn; vi) *sum over all inputs* to suma elementów; vi) *gate activation function (always sigmoid)* to sigmoidalna funkcja

56 L. Medsker, L.C. Jain, *Recurrent Neural Networks: Design and Applications*, CRC Press 1999.
57 S. Hochreiter, J. Schmidhuber, *Long Short-Term Memory*, „Neural Computation" t. 9 nr 8 (1997), DOI: 10.1162/neco.1997.9.8.1735.
58 R. Pascanu, T. Mikolov, Y. Bengio, *On the difficulty of training Recurrent Neural Networks*, arXiv, 15 lutego 2013 r., http://arxiv.org/abs/1211.5063.

aktywacji; vii) *input activation function (usually tanh)* to funkcja aktywacji tangens hiperboliczny; viii) *output activation function (usually tanh)* to funkcja aktywacji tangens hiperboliczny; ix) *input* to aktualne sygnały wejściowe do jednostki LSTM; x) *recurrent* to rekursywne sygnały wejściowe do jednostki LSTM; xi) *peepholes* to dodatkowe, opcjonalne połączenia. Bramki (*gates*) są to sekwencje ważonych iloczynów, sum i aktywacji, a więc neurony, których sygnały wyjściowe krzyżowane są wewnątrz jednostki LSTM za pomocą iloczynów wykonywanych podług połączeń nieważonych (bezpośrednich lub rekursywnych). Istotą jednostki LSTM są operacje składające się na jej komórkę pamięci (*cell*), której stan c utożsamiany jest z pamięcią długotrwałą, bowiem stanowi on informacje przechowywane przez jednostkę i przesyłane na jej własne wyjście. Bramka wejściowa (*input* lub *write gate*) odpowiada za regulowanie dopływu nowych danych do jednostki, może ona bowiem odcinać informacje dochodzące do wejścia jednostki (*block input*), w sytuacjach, gdy dany stan komórki należy zachować. Bramka zapominająca (*forget gate*) odpowiada za regulowanie tego, w jakim stopniu obecny stan komórki należy zapomnieć, aby zastąpić go nowym. Bramka wyjściowa (*output gate*) odpowiada za to, w jakim stopniu udostępnić należy stan komórki do innych jednostek LSTM, reguluje więc wypływ sygnałów wyjściowych z jednostki.

Źródło: https://developer.nvidia.com/discover/lstm, dostęp 13 listopada 2022 r.

Sieci rekurencyjne znajdują najczęstsze zastosowania w przetwarzaniu danych sekwencyjnych lub szeregów czasowych (*e.g.* nagrań audio, wypowiedzi w języku naturalnym, sygnałów poligraficznych, notowań giełdowych, *etc*.) a więc tam, gdzie dane są sekwencjami zależnych od siebie obserwacji.

Sieci konwolucyjne. Jednym z najważniejszych przełomów w rozwoju architektur sieci neuronowych były konwolucyjne sieci neuronowe (*convolutional neural networks*, CNN), które spopularyzowali A. Krizhevsky, I. Sutskever i G. Hinton w 2012 roku[59]. Sieci konwolucyjne stanowiły rozwiązanie problemu przetwarzania danych przestrzennych przez sieci perceptronowe, które wymagały w tym celu ogromnych mocy obliczeniowych. Problem ten wynikał z faktu, że w przypadku obrazów, każdy piksel stanowi element wejściowy do sieci neuronowej, zatem każdy neuron wejściowy potrzebuje tylu synaps, ile jest pikseli na obrazie (*e.g*. obraz o stosunkowo niewielkiej jakości Full HD, rozdzielczość 1920 x 1080 px, składa się z 2073600 px). Inspirowani budową kory wzrokowej ssaków, która umożliwia

59 A. Krizhevsky, I. Sutskever, G.E. Hinton, *ImageNet Classification with Deep Convolutional Neural Networks* [w:] *Advances in Neural Information Processing Systems*, t. 25, Curran Associates, Inc. 2012.

przetwarzanie dużych ilości informacji przestrzennych w krótkim czasie[60]. Kolejni badacze[61] skłaniali się do rezygnacja z podstawowego założenia, głoszącego iż: i) każdy neuron wejściowy potrzebuje tylu synaps co elementów liczą dane wejściowe; ii) każdy neuron potrzebuje tylu synaps ilu ma sąsiadów z którymi się łączy. Podstawową bowiem różnicą pomiędzy siecią perceptronową a konwolucyjną jest to, że pierwsza jest w pełni połączona (*fully connected*) a druga tylko lokalnie (*locally connected*). Rozwiązanie problemu polega na tym, że: i) neurony posiadają ograniczone pola receptywne, których wielkość wyrażana może być w pikselach (*e.g*. pole receptywne 3 x 3 px wymaga 9 połączeń synaptycznych); ii) neuron obserwuje na raz tylko taki fragment obrazu, który mieści się w jego polu receptywnym; iii) neuron przemieszcza swoje pole receptywne po obrazie, a jego kolejne sygnały wyjściowe tworzą nowy obraz, nazywany mapą aktywności (*activation map*); iv) mapy aktywności z danej warstwy neuronów stanowią dane wejściowe do kolejnych warstw.

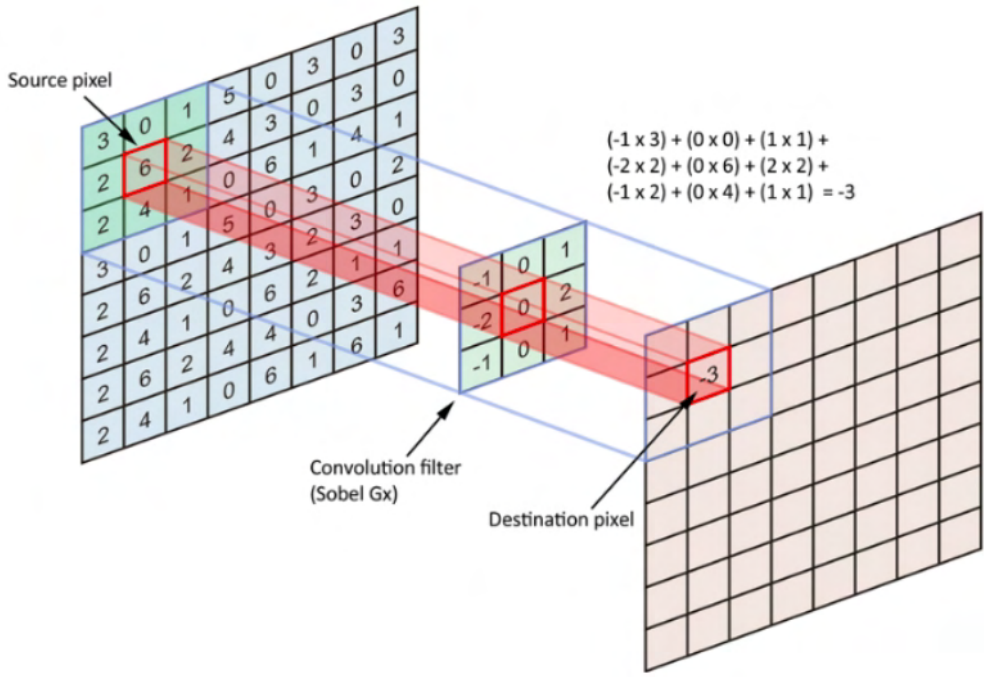
Neurony w sieciach konwolucyjnych nazywane są filtrami (*filter* lub *kernel*), ponieważ, metoda ich działania tożsama jest z metodą działania filtrów stosowanych w obróbce cyfrowej obrazów (*e.g*. filtr gaussowski służy do rozmywania obrazów). Wynikiem filtrowania (nazywanego także konwolucją lub splotem) jest nowy obraz, uzyskiwany poprzez: i) iloczyn pikseli znajdujących się w polu receptywnym filtra z wartościami filtra; ii) zsumowanie wyników iloczynu; iii) przesunięcie filtra.

---

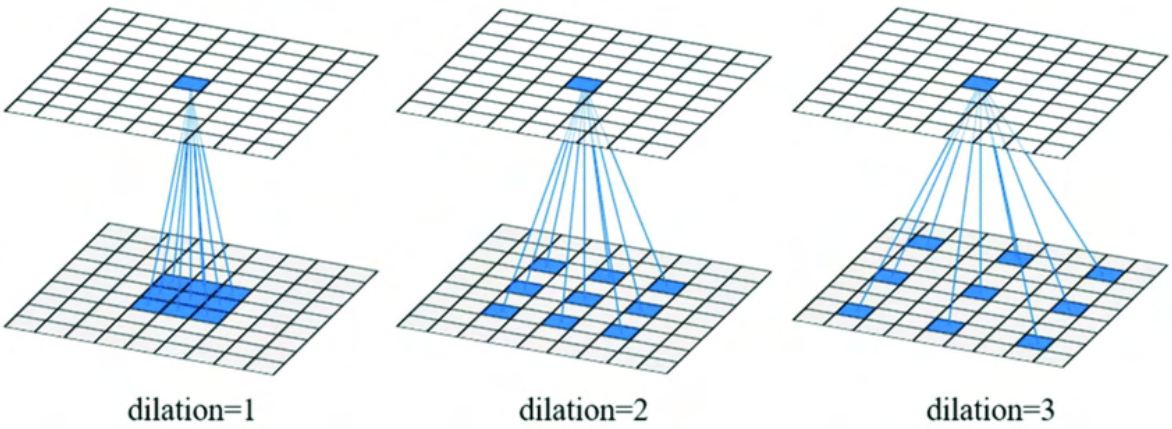
60 D.H. Hubel, T.N. Wiesel, *Receptive fields of single neurones in the cat's striate cortex*, „The Journal of Physiology" t. 148 nr 3 (1959); D.H. Hubel, T.N. Wiesel, *Receptive fields and functional architecture of monkey striate cortex*, „The Journal of Physiology" t. 195 nr 1 (1968), DOI: 10.1113/jphysiol.1968.sp008455.

61 K. Fukushima, *Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position*, „Biological Cybernetics" t. 36 nr 4 (1980), DOI: 10.1007/BF00344251; K. Jarrett, K. Kavukcuoglu, M. Ranzato, Y. LeCun, *What is the best multi-stage architecture for object recognition?* [w:] *2009 IEEE 12th International Conference on Computer Vision*, 2009; A. Krizhevsky, *Convolutional Deep Belief Networks on CIFAR-10*, 2010 r., https://www.semanticscholar.org/paper/Convolutional-Deep-Belief-Networks-on-CIFAR-10-Krizhevsky/bea5780d621e669e8069f05d0f2fc0db9df4b50f; Y. LeCun, B. Boser, J. Denker, D. Henderson, R. Howard, W. Hubbard, L. Jackel, *Handwritten Digit Recognition with a Back-Propagation Network* [w:] *Advances in Neural Information Processing Systems*, t. 2, Morgan-Kaufmann 1989; Y. Lecun, L. Bottou, Y. Bengio, P. Haffner, *Gradient-based learning applied to document recognition*, „Proceedings of the IEEE" t. 86 nr 11 (1998), DOI: 10.1109/5.726791; H. Lee, R. Grosse, R. Ranganath, A.Y. Ng, *Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations* [w:] *Proceedings of the 26th Annual International Conference on Machine Learning*, New York, NY, USA 2009; N. Pinto, D. Doukhan, J.J. DiCarlo, D.D. Cox, *A high-throughput screening approach to discovering good forms of biologically inspired visual representation*, „PLoS computational biology" t. 5 nr 11 (2009), DOI: 10.1371/journal.pcbi.1000579; S.C. Turaga, J.F. Murray, V. Jain, F. Roth, M. Helmstaedter, K. Briggman, W. Denk, H.S. Seung, *Convolutional networks can learn to generate affinity graphs for image segmentation*, „Neural computation" t. 22 nr 2 (2010), DOI: 10.1162/neco.2009.10-08-881.

Definiując warstwę konwolucyjną (rys. 1.4.3) określić należy: i) liczbę neuronów (filtrów); ii) wymiary pól receptywnych (filtrów); iii) dylatację (*dilation*), która umożliwia zwiększenie pola receptywnego, bez zwiększania liczby przetwarzanych na raz danych, poprzez rozproszenie połączeń synaptycznych (rys. 1.4.4); iv) długość kroku (*stride*), czyli liczbę pikseli o jaką przesuwany jest filtr; v) funkcję aktywacji (która odróżnia neurony konwolucyjne od filtrów).



Rysunek 1.4.3. Przykład konwolucji (po lewej obraz wejściowy, po prawej wyjściowy, po środku filtr). Źródło: T.-Y. Lin, P. Goyal, M. Zelenskyy, *Deep Neural Networks in Embedded Systems*. Czech Technical University in Prague 2019, s. 15.



Rysunek 1.4.4. Przykład dylatacji, gdzie połączenia synaptyczne filtra oznaczono kolorem niebieskim. Jest to równoważne dodaniu zerowych połączeń pomiędzy tymi, które rzeczywiście będą aktualizowane. Źródło: X. Cui, K. Zheng, L. Gao, D. Yang, J. Ren, *Multiscale Spatial-Spectral Convolutional Network with Image-Based Framework for Hyperspectral Imagery Classification*, „Remote Sensing" t. 11 (2019).

Ponieważ zadaniem filtrów jest ekstrakcja cech z danych przestrzennych, a na obszarze obrazu, który znajduje się w danym momencie w polu receptywnym, cecha poszukiwana przez dany filtr albo występuje albo nie, to dane wyjściowe z warstw konwolucyjnych nazywane są mapami aktywności cech (dana mapa odnosi się tylko do cechy filtrowanej przez dany neuron).

Rezultatem typowej konwolucji jest liczba map aktywności równa liczbie neuronów danej warstwy. Dotyczy to także warstw kolejnych, na których przetwarzane są mapy aktywności z warstw poprzednich[62]. Wynika to stąd, że filtry konwolucyjne operują na wskroś danych wejściowych (rys. 1.4.5), *i.e.*: i) wysokość i szerokość filtra jest arbitralnie określana podczas definiowania sieci; ii) ale głębokość filtra wynika z liczby map aktywności (lub liczby kanałów obrazu), które będzie on przetwarzał. Na przykład, w przypadku obrazów w skali szarości[63] i obrazów w skali RGB[64]: i) liczby map aktywności będą takie same; ii) ale filtry różnić się będą liczbami połączeń synaptycznych (będą one trzykrotnie wyższe dla obrazów w skali RGB). Jeżeli podczas konwolucji zastosowano łatanie (*padding*), czyli sztucznie poszerzano obraz (na ogół o wartości zerowe), aby filtr mógł operować na jego krawędziach, to mapy aktywności będą miały te same wymiary co obraz wejściowy. W przeciwnym przypadku, rozmiar map aktywności będzie się zmniejszał oraz tracone będą informacje znajdujące się na krawędziach obrazów (rys. 1.4.3). Rozmiar map aktywności zależy także od długości kroku i wartości dylatacji, zarówno krok, jak i dylatacja o wartości większej niż jeden, skutkują mapami aktywności mniejszymi niż obraz wejściowy.

Wymiary map aktywności wyliczyć można następująco (rys. 1.4.6)[65]:

$$O = \frac{I - F + 2P}{S} + 1 \qquad \text{(Równanie 1.4.1)}$$

Gdzie:

O – długość mapy aktywności (na jej wysokości albo szerokości);

62 Jeżeli na warstwie wejściowej jest x neuronów, to po przetworzeniu obrazu, dadzą one x map aktywności, pozornie więc, rezultatem przetworzenia tych map aktywności przez y neuronów warstwy kolejnej, będzie xy map aktywności. Tak jednak nie jest (liczba ich będzie równa y).

63 Skala szarości, gdzie obraz reprezentowany jest przez pojedynczą macierz, której elementy przyjmują wartości w przedziale [0, 255].

64 Skala kolorowa, gdzie obraz reprezentowany jest przez trzy macierze – skala zieleni, czerwieni i niebieskiego – których elementy także przyjmują wartości w przedziale [0, 255].

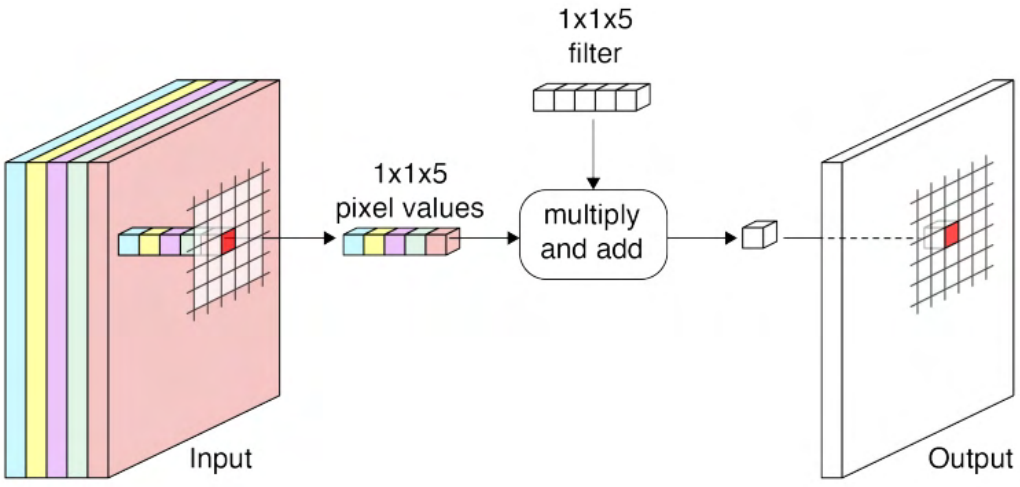
65 A. Amidi, S. Amidi, *Deep Learning; Convolutional Neural Networks. Stanford University* [na:] https://stanford.edu/~shervine/teaching/cs-230/cheatsheet-convolutional-neural-networks, 2019 r., dostęp 12 stycznia 2023 r.

I – długość danych wejściowych;

F – długość filtra;

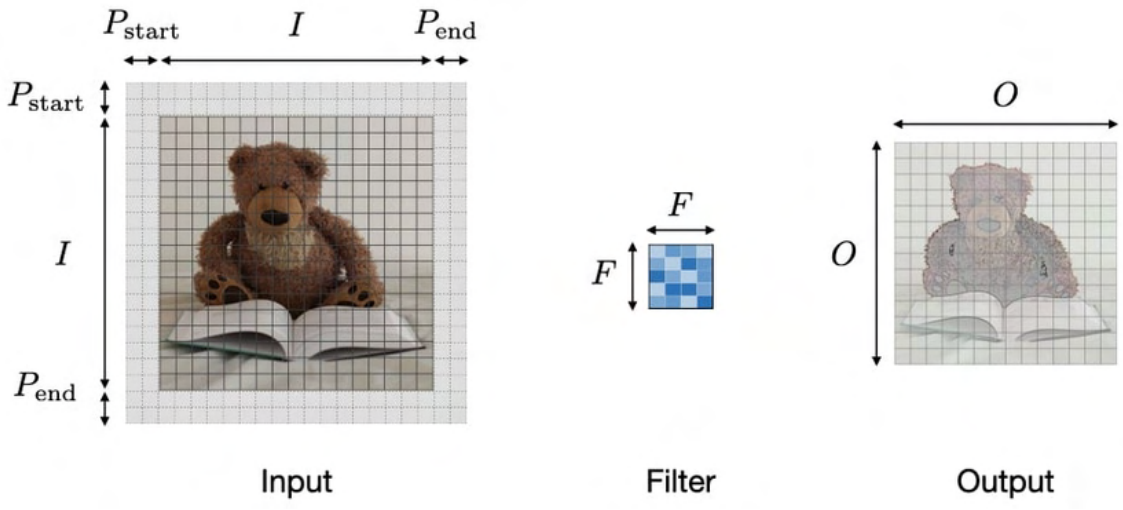
P – długość łatania;

S – krok filtra.



Rysunek 1.4.5. Schemat konwolucji z uwzględnieniem głębokości danych wejściowych (liczby map aktywności lub kanałów obrazu), gdzie: i) *input* to dane wejściowe; ii) *output* to dane wyjściowe; iii) *pixel values* to wartość pikseli; iv) *filter* to filtr; v) *multiply and add* to iloczyn ważony i suma.

Źródło: A. Glassner, *Deep Learning: From Basics to Practice*, t. 2, Seattle 2018, s. 981.



Rysunek 1.4.6. Schemat operacji konwolucji ze względu na wymiary danych wejściowych, filtrów i map aktywności, gdzie: i) $P_{start} = P_{end} = P$; ii) *input* to obraz wejściowy; iii) *filter* to filtr; iv) *output* to dane wyjściowe, czyli mapa aktywności.

Źródło: A. Amidi, S. Amidi, *Deep Learning; Convolutional Neural Networks. Stanford University* [na:] https://stanford.edu/~shervine/teaching/cs-230/cheatsheet-convolutional-neural-networks, 2019 r., dostęp 12 stycznia 2023 r.

Liczbę parametrów danej warstwy uzyskać można z równania[66]:

$$W = (FFC + 1)K \qquad \text{(Równanie 1.4.2)}$$

Gdzie:

W – liczba parametrów danej warstwy kownolucyjnej;

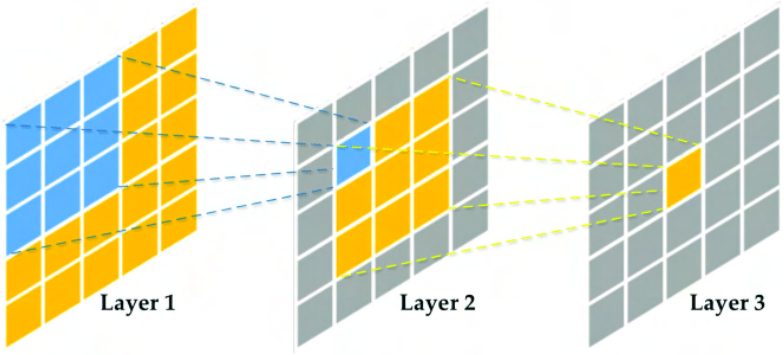
F – długość filtrów;

C – liczba wejściowych map aktywności (lub kanałów na obrazie wejściowym);

K – liczba filtrów na podliczanej warstwie;

1 – przy założeniu jednego parametru *bias* dla każdego filtra.

W przypadku sieci konwolucyjnych podkreśla się, iż są one hierarchiczne, ponieważ kolejne warstwy ekstraktują cechy z map aktywności opisujących cechy wyekstraktowanie już przez warstwy poprzednie. Stąd pola receptywne rosną z warstwy na warstwę (rys. 1.4.7), powodując że: i) warstwy niższe w hierarchii, których efektywne pola receptywne są mniejsze, ekstraktują cechy prostsze; ii) zaś warstwy wyższe w hierarchii, których efektywne pola receptywne są większe, ekstraktują cechy bardziej złożone.

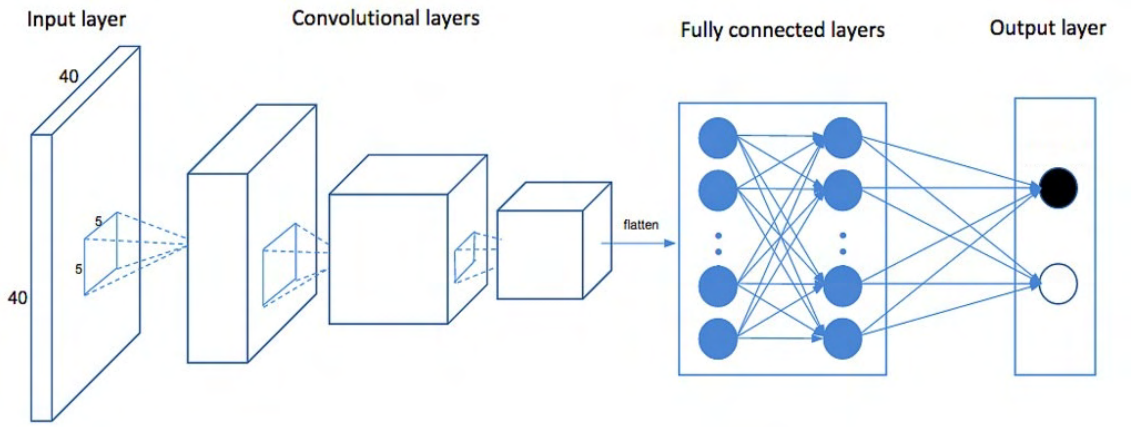


Rysunek 1.4.7. Schemat obrazujący hierarchiczny charakter sieci konwolucyjnych (ze względu na przyrost efektywnych pól receptywnych), gdzie: i) *layer* to warstwa z której pochodzi dana mapa aktywności; ii) kolorem niebieskim oznaczono filtr warstwy drugiej, a żółtym filtr warstwy trzeciej; iii) filtry obydwu warstw mają te same wymiary i pola receptywne, ale efektywne pole receptywne filtra warstwy trzeciej jest znacznie większe, niż filtra warstwy drugiej.

Źródło: P. Xu, Z. Guo, L. Liang, X. Xu, *MSF-Net: Multi-Scale Feature Learning Network for Classification of Surface Defects of Multifarious Sizes*, „Sensors" t. 21 (2021), DOI: 10.3390/s21155125, s. 3.

66 Ibid.

Na ogół sieć konwolucyjna zakończona jest warstwami w pełni połączonymi (perceptronowymi), które przeprowadzają proces decyzyjny na podstawie cech wyekstraktowanych hierarchicznie przez kolejne warstwy konwolucyjne (rys. 1.4.8).



Rysunek 1.4.8. Schemat sieci konwolucyjnej, gdzie: i) *input layer* to warstwa wejściowa; ii) *convolutional layers* to warstwy konwolucyjne; iii) *flatten* to operacja spłaszczenia i konkatenacji map aktywności, aby tworzyły wektor wejściowy do warstw perceptronowych; iv) *fully connected layers* to warstwy w pełni połączone (perceptronowe); v) *output layer* to warstwa wyjściowa.
Źródło: https://blog.insightdatascience.com/automating-breast-cancer-detection-with-deep-learning-d8b49da17950, dostęp 12 stycznia 2023 r.

Głównym zastosowaniem sieci konwolucyjnych jest przetwarzanie danych przestrzennych (na ogół obrazów lub filmów). Wyróżnić tutaj można (rys. 1.4.9): i) klasyfikację obrazów (*image classification*), w której model określa klasy obiektów widocznych na obrazie; ii) wykrywanie obiektów (*object detection*), gdzie model dodatkowo oznacza obiekty, którym przypisuje klasy; iii) segmentację semantyczną (*semantic segmenation*), w której model określa, jakie piksele należą do wykrytych i zaklasyfikowanych obiektów, przy czym obiekty danej klasy ujmowane są zbiorczo; iv) segmentację instancyjną (*instance segmentation*), która tożsama jest z segmentacją semantyczną, przy czym obiekty należące do danej klasy ujmowane są indywidualnie.

(a) Image Classification (b) Object Detection (c) Semantic Segmentation (d) Instance Segmentation

Rysunek 1.4.9. Przykład różnic pomiędzy klasyfikacją obrazów (*image classification*), wykrywaniem obiektów (*object detection*), segmentacją semantyczną (*semantic segmentation*) i segmentacją instancyjną (*instance segmentation*).

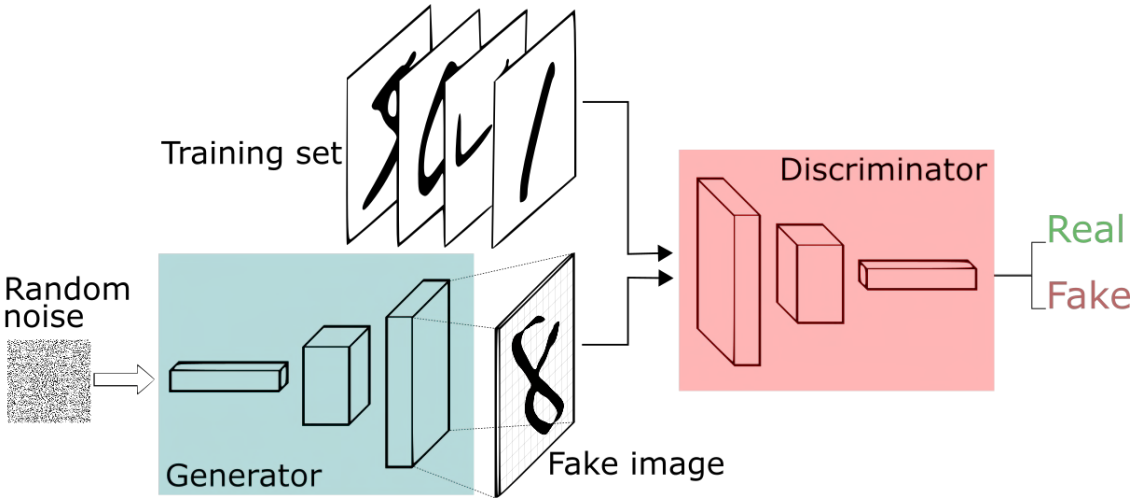
Źródło: X. Wu, D. Sahoo, S.C.H. Hoi, *Recent Advances in Deep Learning for Object Detection*, arXiv, 9 sierpnia 2019 r., http://arxiv.org/abs/1908.03673, s. 1.

Sieci generatywno-adwersaryjne. Jedną z najnowszych architektur sieci neuronowych są modele generatywno-adwersaryjne (*gnerative-adversarial network*, GAN), które po raz pierwszy opublikowali I. Goodfellow *et al.* w 2014 roku[67]. Sieci te opracowano celem generowania danych sztucznych, które byłyby możliwie podobne do naturalnych. Typowa sieć generatywno adwersaryjna składa się z: i) generatora, którego zadaniem jest wygenerować sztuczne dane; ii) oraz dyskryminatora, którego zadaniem jest odróżnić dane sztuczne od prawdziwych. Idea tej metody polega na rywalizacji pomiędzy generatorem i dyskryminatorem, które poprzez wzajemną konkurencję, powinny się wzajemnie od siebie uczyć. Rolę generatora i dyskryminatora pełnią na ogół sieci konwolucyjne (rys. 1.4.10).

Podczas treningu standardowej sieci generatywno-adwersaryjnej: i) na wejścia generatora dostarczany jest losowo generowany szum, który ten przekształca w obraz wyjściowy; ii) na wejścia dyskryminatora dostarczany jest obraz sztuczny lub naturalny, który ten poddaje klasyfikacji; iii) celem generatora jest wytwarzanie obrazów, które dyskryminator uzna za prawdziwe, natomiast celem dyskryminatora jest trafna klasyfikacja przetwarzanych obrazów na sztuczne i prawdziwe. Rywalizacja obydwu modeli polega na tym, że: i) generator, dążąc do minimalizacji (lub maksymalizacji) swojej funkcji kosztu, w wyniku aktualizacji swoich synaps generuje lepsze obrazy sztuczne i powoduje zwiększenie (lub zmniejszenie) kosztu dyskryminatora; ii) a dyskryminator, dążąc do minimalizacji (lub maksymalizacji) swojej funkcji kosztu, w

67 I.J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, *Generative Adversarial Networks*, 10 czerwca 2014 r., http://arxiv.org/abs/1406.2661.

wyniku aktualizacji swoich synaps uczy się lepiej odróżniać obrazy prawdziwe od fałszywych i powoduje zwiększenie (lub zmniejszenie) kosztu generatora.



Rysunek 1.4.10. Schemat sieci generatywno-adwersaryjnej, gdzie: i) *random noise* to losowo wygenerowane dane wejściowe ; ii) *training set* to zbiór obrazów prawdziwych; iii) *fake images* to obrazy sztuczne wytworzone przez generator; iv) *real* i *fake* to klasy prawda i fałsz.
Źródło: https://sthalles.github.io/intro-to-gans/, dostęp 12 stycznia 2023 r.

Przykładowe funkcje kosztu dla generatora i dyskryminatora, które będą maksymalizowane, przedstawić można następująco (funkcja wassersteinowska)[68]:

$$L_G = D(G(x))$$
$$L_D = D(v) - D(G(x))$$

(Równanie 1.4.3)

Gdzie:

L_G – koszt generatora;

L_D – koszt dyskryminatora;

G – funkcja generator;

D – funkcja dyskryminator;

v – dane prawdziwe;

x – dane wejściowe do generatora.

Koszt generatora zależny jest jedynie od tych rozstrzygnięć dyskryminatora, które dotyczą wygenerowanych przezeń obrazów. Natomiast, koszt dyskryminatora zależny

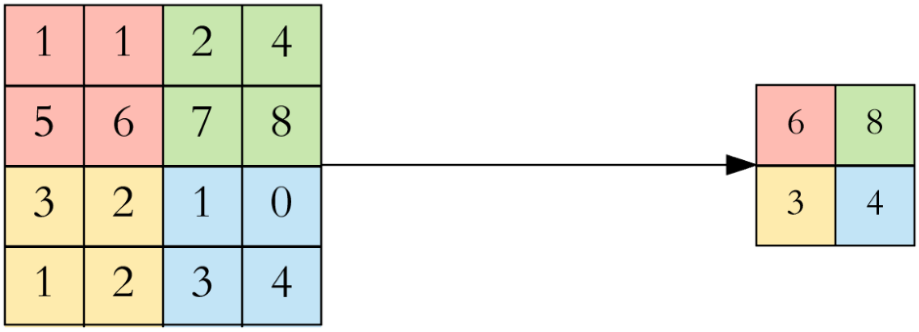
68 M. Lucic, K. Kurach, M. Michalski, S. Gelly, O. Bousquet, *Are GANs Created Equal? A Large-Scale Study*, arXiv, 29 października 2018 r., http://arxiv.org/abs/1711.10337, s. 3; L. Weng, *From GAN to WGAN*, arXiv, 18 kwietnia 2019 r., http://arxiv.org/abs/1904.08994, s. 11.

jest od wszystkich rozstrzygnięć, które podjął. Celem generatora jest w tym przypadku maksymalizacja sygnałów wyjściowych z dyskryminatora, zaś celem dyskryminatora jest tutaj zwiększenie różnicy między jego sygnałami wyjściowymi dla obrazów sztucznych i naturalnych.

Najważniejszym zastosowaniem sieci generatywno-adwersaryjnych jest wytwarzanie nowych danych na podstawie danych istniejących. Na ogół stosuje się sieci oparte o modele konwolucyjne, dla wytwarzania nowych danych przestrzennych.

1.5. Warstwy sztucznych sieci neuronowych.

Warstwy redukujące. Ponieważ sieci konwolucyjne przetwarzają znacznie ilości danych w postaci map aktywności, to pożądaną jest kontrola nad ich rozdzielczością, do czego służą tzw. warstwy redukujące (*downsampling* lub *pooling layers*), umieszczane pomiędzy warstwami konwolucyjnymi.



Rysunek 1.5.1. Schemat warstwy redukującej przez wyciągnięcie najwyższej wartości znajdującej się w oknie (*max-pooling layer*), gdzie rozmiar okna wynosi 2 x 2 px, a długość kroku 2 px.
Źródło: T.-Y. Lin, P. Goyal, M. Zelenskyy, *Deep Neural Networks in Embedded Systems*. Czech Technical University in Prague 2019, s. 17.

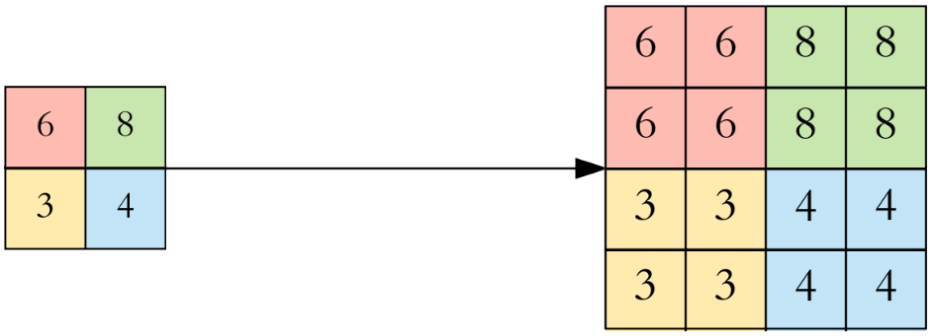
Typowa warstwa redukująca posiada okno, które w sposób podobny do działania filtra, przemierza mapę aktywności z określonym krokiem (*stride*). Standardowy wariant warstwy redukującej wyciąga najwyższą spośród wartości znajdujących się w oknie (rys. 1.5.1). Inne warianty warstw redukujących polegają na zsumowaniu lub wyciągnięciu średniej z wartości znajdujących się w oknie.

Wariant globalny warstw redukujących polega na wyciągnięciu najwyższej wartości, średniej lub sumy z całej mapy aktywności. Warstwy redukujące globalnie

znajdują najczęstsze zastosowanie pomiędzy ostatnią warstwą konwolucyjną, a pierwszą warstwą w pełni połączoną.

Warstwy zwiększające. Częste zastosowanie znajdują też warstwy zwiększające rozdzielczość (*upsamling layer*), szczególnie w sieciach generatywno-adwersaryjnych, gdzie generator takiej sieci otrzymuje na wejście dane o mniejszej rozdzielczości, niż na ogół oczekiwana na jego wyjściu.

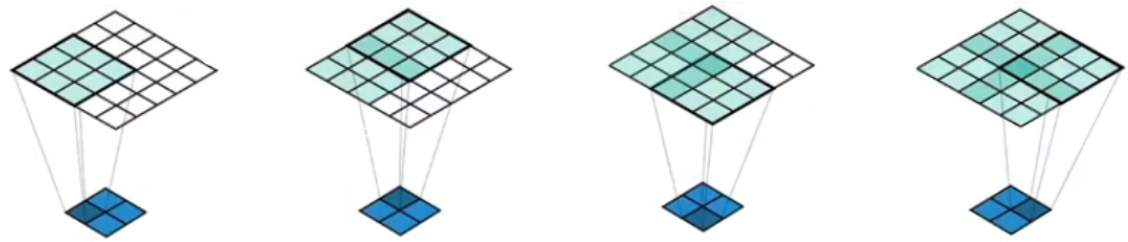
Na przykład, w przypadku warstwy zwiększającej rozdzielczość metodą najbliższych sąsiadów (*nearest neighbor upsampling*): i) okno, od strony danych wejściowych (*i.e.* przemierzając mapę aktywności), ma wymiary 1 x 1 px i krok długości 1; ii) zaś od strony wyjściowej (*i.e.* przemierzając tworzoną mapę), ma wymiary arbitralnie określone, które wskazują ilekroć powiększona zostanie mapa wejściowa (krok równy jest wymiarom okna); iii) okno, przemierzając wejściową mapę aktywności, kopiuje kolejne jej wartości do mapy wyjściowej, gdzie dany piksel kopiowany jest tyle razy, ile wynika z wyjściowych wymiarów okna (rys. 1.5.2).



Rysunek 1.5.2. Schemat warstwy zwiększającej wymiary mapy aktywności (*upsampling layer*) przez klonowanie wartości z mapy istniejącej, gdzie rozmiar okna i długość kroku wynoszą 2 x 2 px.
Źródło: T.-Y. Lin, P. Goyal, M. Zelenskyy, *Deep Neural Networks in Embedded Systems*. Czech Technical University in Prague 2019, s. 17.

Transponowana konwolucja. Zastosowanie warstw redukujących lub zwiększających rozdzielczość nie zawsze jest pożądane, szczególnie w sieciach generatywno-adwersaryjnych. Gdzie, preferowaną często metodą jest zmniejszanie lub zwiększanie wymiarów map aktywności za pomocą długości kroku stosowanego podczas konwolucji i wielkości filtra stosowanego podczas transponowanej konwolucji.

Konwolucja transponowana (*transposed convolution*), nazywana też bywa dekonwolucją (*deconvolution*), pomimo iż nie jest to proces odwrotny do konwolucji (jej inwersja). Podczas transponowanej konwolucji (rys. 1.5.3): i) filtr, od strony wejściowej, ma wymiary 1 x 1 px i krok 1; ii) zaś od strony wyjściowej, ma wymiary i krok arbitralnie określone, które wyznaczać będą o ile zwiększony zostanie rozmiar wyjściowej mapy aktywności; iii) wykonywany jest iloczyn piksela wejściowego z elementami filtra (tym samym, z jednego piksela uzyskuje się macierz pikseli o wymiarach równych wymiarom filtra); iv) jeżeli sygnały wyjściowe uzyskiwane podczas kolejnych kroków nakładają się na siebie, ponieważ krok filtra od strony wyjściowej jest mniejszy niż jego rozmiar (*e.g.* przy kroku równym 2 px uzyskiwana jest macierz o wymiarach 3 x 3 px), to sumowane są tam gdzie się nakładają.



Rysunek 1.5.3. Schemat transponowanej konwolucji, gdzie: i) wejściowa mapa aktywności znajduje się na dole; ii) wyjściowa mapa aktywności znajduje się na górze; iii) filtr ma wymiary 3 x 3 px; iv) długość kroku na wyjściu wynosi 2 px; iv) a nakładające się na siebie sygnały są sumowane.
Źródło: https://medium.com/apache-mxnet/transposed-convolutions-explained-with-ms-excel-52d13030c7e8, dostęp 13 stycznia 2023 r.

Wymiary map aktywności po zastosowaniu transponowanej konwolucji wyliczyć można następująco (rys. 1.5.4):

$$O = S(I - 1) + F - 2P \qquad \text{(Równanie 1.5.1)}$$

Gdzie:

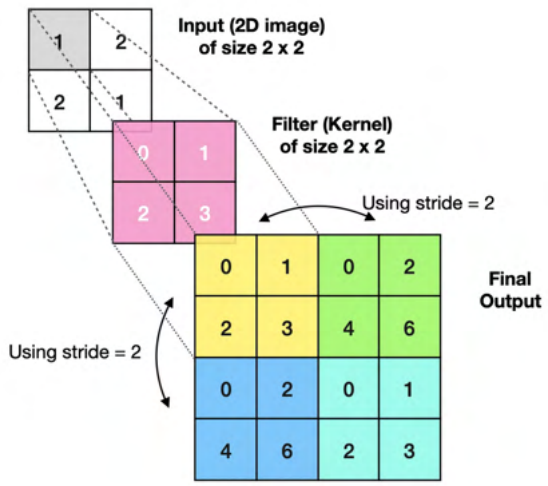
O – długość mapy aktywności (na jej wysokości albo szerokości);

I – długość danych wejściowych;

F – długość filtra;

P – długość łatania;

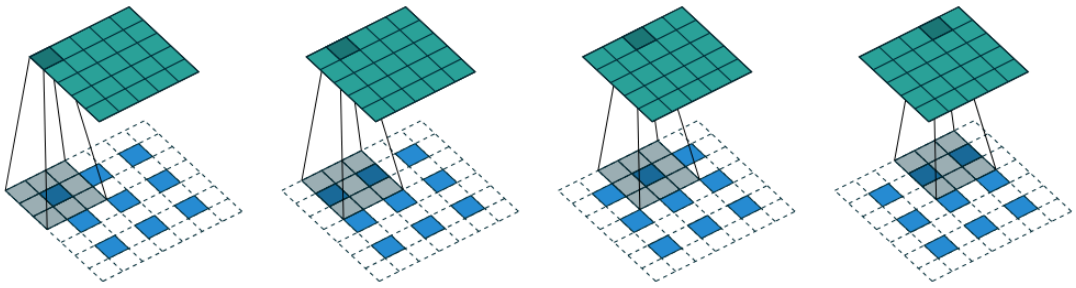
S – krok filtra.



Rysunek 1.5.4. Schemat transponowanej konwolucji, gdzie: i) *input* to wejściowa mapa aktywności o wymiarach 2 x 2 px; ii) *kernel* to filtr o wymiarach 2 x 2 px; iii) *stride* to długość kroku filtra na wyjściu, która wynosi 2 px; iv) *final output* to mapa wyjściowa o wymiarach 4 x 4 px.

Źródło: https://towardsdatascience.com/transposed-convolutional-neural-networks-how-to-increase-the-resolution-of-your-image-d1ec27700c6a, dostęp 13 stycznia 2023 r.

W praktyce transponowana konwolucja implementowana jest jednak odmiennie, co nazwać można krokową konwolucją transponowaną (*strided transposed convolution*)[69]. Powodem dla zastosowania tej metody jest jej prostota, stosuje się bowiem zwyczajną konwolucję (rys. 1.5.5), ale piksele wejściowej mapy aktywności rozpraszane są poprzez umieszczanie pomiędzy nimi pikseli o wartości zerowej (są też łatane na brzegach pikselami o zerowej wartości).



Rysunek 1.5.5. Schemat krokowej transponowanej konwolucji, gdzie: i) dane wejściowe znajdują się na dole; ii) dane wyjściowe znajdują się na górze; iii) zastosowano filtr o wymiarach 3 x 3 px; iv) długość kroku wynosi 1 px.

Źródło: V. Dumoulin, F. Visin, *A guide to convolution arithmetic for deep learning*, arXiv, 11 stycznia 2018 r., http://arxiv.org/abs/1603.07285, s. 25.

69 V. Dumoulin, F. Visin, *A guide to convolution arithmetic for deep learning*, arXiv, 11 stycznia 2018 r., http://arxiv.org/abs/1603.07285.

Warstwa opuszczająca. Jedną z najbardziej popularnych metod pozwalających na uniknięcie przeuczenia sieci są warstwy opuszczające (*dropout layer*)[70], umieszczane arbitralnie pomiędzy warstwami neuronów. Warstwy te w każdym kroku uczenia wyzerowują losowo wybrane połączenia synaptyczne pomiędzy warstwami neuronów, utrudniając sieci naukę poprzez zapamiętywanie danych. Podstawowym parametrem, który określić należy dla takiej warstwy, jest prawdopodobieństwo z jakim dana synapsa może zostać w danym kroku zignorowana (rozumieć to także można jako odsetek połączeń, które zostaną odłączone).

Warstwa normalizująca. Jedną z najbardziej popularnych metod, które wpływają na poprawę stabilności i znaczne przyśpieszenie procesów uczenia sieci neuronowych, są warstwy normalizujące wartość sygnałów przesyłanych pomiędzy warstwami neuronowymi (*normalization layer*). W 2015 roku S. Ioffe i C. Szegedy zaproponowali[71], aby wykorzystywać takie warstwy, które normalizować będą sygnały ze względu na wszystkie przykłady, jakie zostały zaprezentowane podczas danego kroku uczenia. Jest to tak zwana *batch-normalization layer*, która stanowi najbardziej powszechną obecnie metodę normalizacji w sieciach neuronowych.

Podczas prezentacji kolejnych przykładów, zakresy wartości sygnałów przesyłanych pomiędzy kolejnymi warstwami neuronów mogą znacznie się różnić, co wpływać będzie negatywnie na czas i stabilność uczenia. W tym celu normalizuje się sygnały neuronów, zapewniając że będą mieć średnią μ bliską 0 i odchylenie standardowe σ bliskie 1, a następnie umożliwia się sieci korektę średniej i odchylenia standardowego za pomocą parametrów β i γ, których sieć może się nauczyć.

Normalizację przeprowadza się w każdym kroku uczenia. Krok uczenia polega na przetworzeniu przykładów i aktualizacji połączeń synaptycznych ze względu na te przykłady. Długość kroku uczenia określa ilość przykładów, ze względu na które aktualizowane będą połączenia synaptyczne (na ogół aktualizuje się połączenia synaptyczne ze względu na wiele przykładów równocześnie). Podzbiór przykładów, które obejmowane są jednym krokiem uczenia, określa się jako *batch* (serię), długość kroku uczenia jako *batch-size*, a normalizację *batch-normalization.*

70 P. Baldi, P.J. Sadowski, *Understanding Dropout* [w:] *Advances in Neural Information Processing Systems*, t. 26, Curran Associates, Inc. 2013.

71 S. Ioffe, C. Szegedy, *Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift*, „arXiv:1502.03167 [cs]" (2015), http://arxiv.org/abs/1502.03167.

Zakładając, że: i) dana warstwa liczy n neuronów (gdzie $i = 1, 2, \ldots, n$); ii) a krok uczenia liczy k przykładów (gdzie $j = 1, 2, \ldots, k$). To *batch* sygnałów wyjściowych uzyskanych z i-tego neuronu podczas całego kroku ująć można jako:

$$B_i = [y_1, y_2, \ldots, y_k] \qquad \text{(Równanie 1.5.2)}$$

Średnią[72] μ sygnałów i-tego neuronu podczas całego kroku oblicza się jako[73]:

$$\mu = \frac{1}{k}\sum_{j=1}^{k} y_j \qquad \text{(Równanie 1.5.3)}$$

Odchylenie standardowe[74] σ sygnałów i-tego neuronu podczas całego kroku dane jest równaniem[75]:

$$\sigma = \sqrt{\frac{1}{k}\sum_{j=1}^{k}(y_j - \mu)^2} \qquad \text{(Równanie 1.5.4)}$$

Normalizacja[76] j-tego sygnału z i-tego neuronu ze względu na cały krok uczenia określona zostanie jako[77]:

$$\hat{y}_j = \frac{y_j - \mu}{\sigma} \qquad \text{(Równanie 1.5.5)}$$

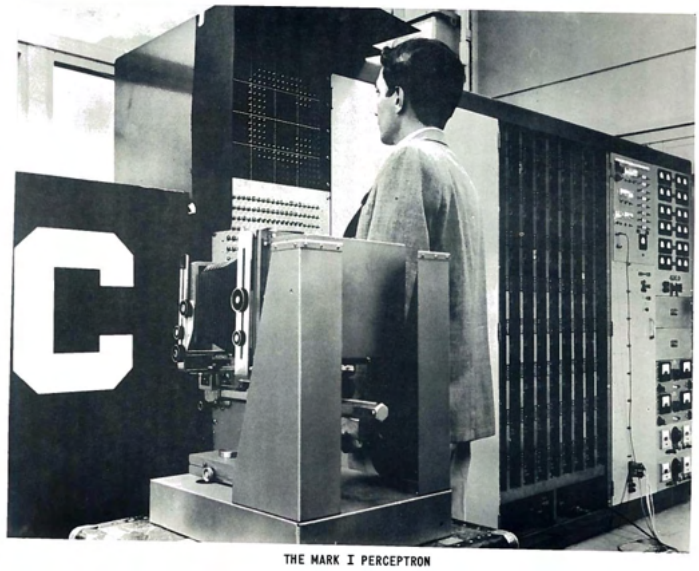
Korekta znormalizowanego j-tego sygnału z i-tego neuronu, przeprowadzana za pomocą parametrów β i γ (których sieć sama się uczy), dana będzie jako[78]:

$$s_j = \gamma\hat{y}_j + \beta \qquad \text{(Równanie 1.5.6)}$$

Dopiero sygnał s_j, który został znormalizowany i skorygowany, zostanie przesłany na neurony warstwy kolejnej.

72 Formalnie określana jako pierwszy moment, jest to średnia odległość od zera.
73 S. Ioffe, C. Szegedy, *Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift*, „arXiv:1502.03167 [cs]" (2015), http://arxiv.org/abs/1502.03167, s. 3.
74 Formalnie określana jako drugi moment, jest to średnia odległość od średniej.
75 S. Ioffe, C. Szegedy, *Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift*, „arXiv:1502.03167 [cs]" (2015), http://arxiv.org/abs/1502.03167, s. 3.
76 Określana też jako standaryzacja Z, zapewnia że średnia bliska będzie 0, a odchylenie standardowe 1.
77 S. Ioffe, C. Szegedy, *Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift*, „arXiv:1502.03167 [cs]" (2015), http://arxiv.org/abs/1502.03167, s. 3.
78 Ibid.

1.6. Historia sztucznych sieci neuronowych. Pierwszą sztuczną sieć neuronową, *Perceptron* (rys. 1.6.1), opracował w 1957 roku F. Rosenblatt[79]. Początkowo było to urządzenie elektromechaniczne[80], ale już w latach 1960/1964 perceptrony były symulowane dla potrzeb CIA (*Central Intelligence Agency*) na komputerach IBM 704[81].



Rysunek 1.6.1. Fotografia pierwszego perceptronu (*Perceptron Mark I*), który uczony był rozpoznawania znaków alfanumerycznych (*e.g.* litera C w lewym dolnym rogu zdjęcia).
Źródło: F. Rosenblatt, *Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms*, Buffalo 1961, s. iii.

Pomimo początkowego zainteresowania i postępów w dziedzinie sztucznych sieci neuronowych, uległa ona wkrótce zapomnieniu. Między innymi za sprawą monografii, którą opublikowali M. Minsky i S. Papert w 1969 roku[82], podważając możliwość rozwiązywania nawet prostych funkcji logicznych za pomocą sieci neuronowych. Następujące dekady, podczas których wiele podmiotów zaprzestało finansowania badań nad sieciami neuronowymi, zyskały miano „zimy sztucznej inteligencji" (*AI winter*).

Pomimo tego, w kolejnych latach czyniono istotne postępy. Przede wszystkim, w 1981 roku P. Werbos jako pierwszy[83] zaproponował zastosowanie propagacji

79 F. Rosenblatt, *The perceptron: A probabilistic model for information storage and organization in the brain*, „Psychological Review" t. 65 (1958), DOI: 10.1037/h0042519.
80 R. Tadeusiewicz, *Sieci Neuronowe*, Warszawa 1993, s. 8.
81 T. Babcock, G. Richmond, M. Spooner, W. Holmes, *Application of Perceptrons to Photointerpretation*, Buffalo 1964.
82 M. Minsky, S. Papert, *Perceptrons; an Introduction to Computational Geometry*, MIT Press 1969.
83 J. Schmidhuber, *Deep Learning in Neural Networks: An Overview*, „Neural Networks" t. 61 (2015), DOI: 10.1016/j.neunet.2014.09.003, s. 11–12.

wstecznej celem uczenia nieliniowych sieci wielowarstwowych[84]. Propozycja ta spopularyzowana została jednak dopiero w 1986 roku za sprawą publikacji D. Rumelharta, R. Williamsa i G. Hintona w czasopiśmie *Nature*[85].

Za sprawą powyższego artykułu spopularyzowano również rekurencyjne sieci neuronowe, których szczególną postać zaproponował jako pierwszy J. Hopfield w 1982 roku (tzw. sieć Hopfielda)[86]. W 1997 roku S. Hochreiter i J. Schmidhuber opracowali sieć typu *Long Short-Term Memory*[87].

W 1980 roku K. Fukushima opublikował *Neocognitron*[88], który stanowi protoplastę sieci konwolucyjnych. Jednym z ważniejszych etapów w rozwoju sieci konwolucyjnych był model *LeNet-5*, który opracowali Y. LeCun *et al*. w 1998 roku[89]. Zasługa spopularyzowania sieci konwolucyjnych przypadła zespołowi A. Krizhevsky, I. Sutskever i G. Hinton, za sprawą artykułu z 2012 roku[90]. Sieci generatywno-adwersaryjne opracował i spopularyzowali zaś I. Goodfellow *et al.* w 2014 roku[91].

Wspomniani powyżej współwynalazcy i popularyzatorzy propagacji wstecznej, sieci konwolucyjnych i sieci generatywno-adwersaryjnych, *etc.*, a mianowicie G. Hinton, Y. LeCun i Y. Bengio, uhonorowani zostali Nagrodą Turinga (*Alan Mathison Turing Award*) za rok 2018. Przyznana im została między innymi za: „[...] konceptualne i inżynierskie przełomy, które uczyniły głębokie sieci neuronowe krytycznym komponentem [nauk] obliczeniowych”[tłum. własne][92]. Nagroda Turinga przyznawana jest przez *Association for Computing Machinery*, a bywa często określana mianem „informatycznej nagrody Nobla”.

84 P.J. Werbos, *Applications of advances in nonlinear sensitivity analysis* [w:] R. F. Drenick, F. Kozin (red.), *System Modeling and Optimization*, Berlin, Heidelberg 1982.

85 D.E. Rumelhart, G.E. Hinton, R.J. Williams, *Learning representations by back-propagating errors*, „Nature” t. 323 nr 6088 (1986), DOI: 10.1038/323533a0.

86 J.J. Hopfield, *Neural networks and physical systems with emergent collective computational abilities.*, „Proceedings of the National Academy of Sciences” t. 79 nr 8 (1982), DOI: 10.1073/pnas.79.8.2554.

87 S. Hochreiter, J. Schmidhuber, *Long Short-Term Memory*, „Neural Computation” t. 9 nr 8 (1997), DOI: 10.1162/neco.1997.9.8.1735.

88 K. Fukushima, *Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position*, „Biological Cybernetics” t. 36 nr 4 (1980), DOI: 10.1007/BF00344251.

89 Y. Lecun, L. Bottou, Y. Bengio, P. Haffner, *Gradient-based learning applied to document recognition*, „Proceedings of the IEEE” t. 86 nr 11 (1998), DOI: 10.1109/5.726791.

90 A. Krizhevsky, I. Sutskever, G.E. Hinton, *ImageNet Classification with Deep Convolutional Neural Networks* [w:] *Advances in Neural Information Processing Systems*, t. 25, Curran Associates, Inc. 2012.

91 I.J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, *Generative Adversarial Networks*, 10 czerwca 2014 r., http://arxiv.org/abs/1406.2661.

92 J. Ormond, *Fathers of the Deep Learning revolution receive 2018 ACM A.M. Turing Award* [na:] https://www.acm.org/media-center/2019/march/turing-award-2018, dostęp 27 stycznia 2023 r.

Rozdział 2. Sztuczne sieci neuronowe w kryminalistyce i antykryminalistyce.

2.1. Kryminalistyka obliczeniowa. Pojęcie kryminalistyki obliczeniowej (*computational forensics*) bywa różnie rozumiane, a wyodrębnienie tej dziedziny nie zostało jeszcze powszechnie zaakceptowane. W opinii autora, konsensualnym mógłby być podział na: i) badania kryminalistyczne przeprowadzane z wykorzystaniem metod obliczeniowych (*e.g*. sztucznych sieci neuronowych), które klasyfikowane będą według dziedziny wykonywanych badań (*e.g.* daktyloskopia); ii) oraz badania zmierzające do opracowania lub przystosowania metod obliczeniowych dla celów badań kryminalistycznych, które określane będą mianem kryminalistyki obliczeniowej. Chociaż przyporządkowanie kryminalistyki obliczeniowej do dziedziny informatyki kryminalistycznej jest zasadne, to zależnie od danej definicji informatyki kryminalistycznej (*e.g*. dziedzina zajmująca się analizą dowodów cyfrowych)[93], definicja kryminalistyki obliczeniowej może odbiegać od zaproponowanej powyżej i powodować niejasności.

Postulat wyróżnienia kryminalistyki obliczeniowej przedstawiony został w związku z pierwszym Międzynarodowym Warsztatem Kryminalistyki Obliczeniowej (*International Workshop on Computational Forensics*, IWCF), który odbył się w 2007 roku (Manchester)[94]. Jak argumentowali współorganizatorzy, S. Srihari i K. Franke, w sprawozdaniu z drugiej IWCF (rok 2008, Washington): „Kilka wyrażeń używanych jest obecnie dla określenia matematycznego i obliczeniowego podejścia do kryminalistyki. *Statystyka Kryminalistyczna* i *Informatyczne Technologie Kryminalistyczne* posiadają najdłuższą tradycję, są jednak specyficzne. Określenia *Inteligencja Kryminalistyczna* i *Kryminalistyka Obliczeniowa* posiadają szersze spektrum. Jest koniecznym, aby ustalić solidne ramy pojęciowe dla *Kryminalistyki Obliczeniowej*, tak jak miało to miejsce w przypadku [innych nauk obliczeniowych]. [...] *Kryminalistyka Obliczeniowa* zmierzać powinna do

– lepszego rozumowania o kryminalistyce,

– ewaluacji podstaw poszczególnych metod naukowych, i

93 E. Gruza, M. Goc, J. Moszczyński, *Kryminalistyka. Czyli o współczesnych metodach dowodzenia przestępstw*, Warszawa 2020, s. 695.

94 N. Zhang, A. Abraham (red.), *Proceedings of the Third International Symposium on Information Assurance and Security, IAS 2007, August 29-31, 2007, Manchester, United Kingdom*, IEEE Computer Society 2007, s. 383–446.

– systematycznego ujęcia kryminalistyki poprzez zastosowanie technik informatycznych, stosowanej matematyki i statystyki.”[95].

Wśród technik obliczeniowych, których opracowanie dla badań kryminalistycznych denotuje kryminalistykę obliczeniową, autorzy sprawozdania wskazali: i) procesowanie sygnałów i obrazów, oraz wizualizację danych, które służyć mogą redukcji przestrzenności danych dla lepszego rozumienia ich przez ekspertów, lub dla wprowadzenia danych do modelu uczenia maszynowego; ii) automatyczne rozpoznawanie obiektów wizualnych; iii) klasyfikację danych statystycznych, która pomóc może w interpretacji pomiarów i określaniu prawdopodobieństw; iv) eksplorację danych (*data mining*), na przykład poprzez klasteryzację, w celu odkrycia nieznanych dotąd zależności między określonymi obiektami; v) robotykę, która przykładowo służyć może replikowaniu ludzkich ruchów przez maszyny, gwarantując ich porównywalności; vi) uczenie maszynowe, którego głównym narzędziem są sieci neuronowe. Argumentowano także, że: „Znaczna część metod obliczeniowej / maszynowej inteligencji zdominowana jest przez algorytmy bazujące na statystyce. Metody te idealnie nadają się do wykorzystania w kryminalistyce, tam gdzie określić należy poziom błędów i obliczyć prawdopodobieństwa”[96].

Autorzy sprawozdania sformułowali też cele i warunki, których realizacja umożliwić powinna, ich zdaniem, dalszy rozwój kryminalistyki obliczeniowej:

> ”i) zwiększenia świadomości o [możliwym] wpływie narzędzi komputerowych na zapobieganie i ściganie przestępstw; z jednej strony pośród kryminalistyków, którzy specjalizują się *e.g.* w biologii, chemii lub medycynie, ale posiadają ograniczone doświadczenie w naukach obliczeniowych, i z drugiej strony, pośród naukowców obliczeniowych, którzy nieświadomi są trudności jakie przedstawia dziedzina [kryminalistyki],
>
> ii) zapoznania naukowców obliczeniowych z potrzebami, procedurami i technikami kryminalistyki, oraz
>
> iii) spowodowania badań nad kryminalistycznymi narzędziami obliczeniowymi i zachęcenia kryminalistyków, oraz naukowców obliczeniowych do wspólnych prac [nad tymi technikami]

95 S.N. Srihari, K. Franke (red.), *Computational Forensics: Second International Workshop, IWCF 2008, Washington, DC, USA, August 7-8, 2008. Proceedings*, t. 5158, Berlin, Heidelberg 2008, s. 4.
96 Ibid., s. 7.

[...] Systematyczne podejście do *Kryminalistyki Obliczeniowej* zapewni wszechstronność badań, rozwój i proces badawczy, który pozostanie skupiony na potrzebach i problemach kryminalistyki”[97].

2.2. Zastosowania sztucznych sieci neuronowych w kryminalistyce. Aktualnie trudno byłoby wskazać dziedzinę, gdzie sieci neuronowe nie znalazły jeszcze zastosowania lub w której nie trwają badania nad zastosowaniem sieci neuronowych.

Na co dzień ludzie mają kontakt z modelami uczenia maszynowego stosowanymi między innymi do tłumaczenia tekstów z różnych języków[98], do rozpoznawania mowy i generowania na tej podstawie tekstu lub napisów[99], bywają one przeciwnikami szachowymi[100], prowadzą autonomiczne samochody[101], rozpoznają pismo odręczne[102] lub dźwięk towarzyszący jego tworzeniu[103] i przekładają je na pismo maszynowe, rozpoznają twarze odblokowując telefony komórkowe[104], filtrują spam[105] i określają zachowania niezgodne z regulaminami platform społecznościowych[106], pomagają w diagnostyce medycznej[107], analizują dane osobowe przechwytywane w

97 Ibid., s. 5–8.

98 D. Castelvecchi, *Deep learning boosts Google Translate tool*, „Nature” (2016), DOI: 10.1038/nature.2016.20696, https://www.nature.com/articles/nature.2016.20696.

99 Z. Zhang, J. Geiger, J. Pohjalainen, A.E.-D. Mousa, W. Jin, B. Schuller, *Deep Learning for Environmentally Robust Speech Recognition: An Overview of Recent Developments*, 21 września 2018 r., http://arxiv.org/abs/1705.10874.

100 D. Silver, T. Hubert, J. Schrittwieser, I. Antonoglou, M. Lai, A. Guez, M. Lanctot, L. Sifre, D. Kumaran, T. Graepel, T. Lillicrap, K. Simonyan, D. Hassabis, *Mastering Chess and Shogi by Self-Play with a General Reinforcement Learning Algorithm*, „arXiv:1712.01815 [cs]” (2017), http://arxiv.org/abs/1712.01815.

101 Q. Rao, J. Frtunikj, *Deep learning for self-driving cars: chances and challenges* [w:] *Proceedings of the 1st International Workshop on Software Engineering for AI in Autonomous Systems*, New York, NY, USA 2018.

102 J. Pastor-Pellicer, M.J. Castro-Bleda, S. España-Boquera, F. Zamora-Martínez, *Handwriting recognition by using deep learning to extract meaningful features*, „AI Communications” t. 32 nr 2 (2019), DOI: 10.3233/AIC-170562.

103 H. Du, P. Li, H. Zhou, W. Gong, G. Luo, P. Yang, *WordRecorder: Accurate Acoustic-based Handwriting Recognition Using Deep Learning* [w:] *IEEE INFOCOM 2018 - IEEE Conference on Computer Communications*, 2018.

104 F. Schroff, D. Kalenichenko, J. Philbin, *FaceNet: A Unified Embedding for Face Recognition and Clustering* [w:] *W: Conference on Computer Vision and Pattern Recognition. IEEE, Piscataway*, 2015.

105 I. AbdulNabi, Q. Yaseen, *Spam Email Detection Using Deep Learning Techniques*, „Procedia Computer Science” t. 184 (2021), DOI: 10.1016/j.procs.2021.03.107.

106 W.-F. Chen, L.-W. Ku, *UTCNN: a Deep Learning Model of Stance Classificationon on Social Media Text*, „arXiv:1611.03599 [cs]” (2016), http://arxiv.org/abs/1611.03599.

107 M. Bakator, D. Radosav, *Deep Learning and Medical Diagnosis: A Review of Literature*, „Multimodal Technologies and Interaction” t. 2 nr 3 (2018), DOI: 10.3390/mti2030047.

internecie i personalizują wyświetlane reklamy[108] lub proponowane informacje[109], identyfikują ludzi na ulicach[110], udzielają informacji na infoliniach[111], doradzają maklerom bankowym[112], koloryzują[113] i udźwiękawiają[114] filmy, *etc*.

Podobnie częste są ich zastosowania w nauce i dziedzinach stosujących naukę, gdzie sieci neuronowe służą między innymi do identyfikacji lezji mózgu[115], przewidywania trójwymiarowych struktur białkowych[116], do badań patologicznych poświęconych wielu trudnym lub rzadkim chorobom[117], do przewidywania rozprzestrzeniania się chorób zakaźnych[118], do określania warunków dla przeprowadzania procesów metalurgicznych[119], do wczesnego wykrywania pożarów

108 M. Ali, P. Sapiezynski, A. Korolova, A. Mislove, A. Rieke, *Ad Delivery Algorithms: The Hidden Arbiters of Political Messaging*, „arXiv:1912.04255 [cs]" (2019), http://arxiv.org/abs/1912.04255.

109 M. Ali, P. Sapiezynski, M. Bogen, A. Korolova, A. Mislove, A. Rieke, *Discrimination through optimization: How Facebook's ad delivery can lead to skewed outcomes*, „Proceedings of the ACM on Human-Computer Interaction" t. 3 nr CSCW (2019), DOI: 10.1145/3359301.

110 D.S. Trigueros, L. Meng, M. Hartnett, *Face Recognition: From Traditional to Deep Learning Methods*, „arXiv:1811.00116 [cs]" (2018), http://arxiv.org/abs/1811.00116.

111 R. Csaky, *Deep Learning Based Chatbot Models*, „arXiv:1908.08835 [cs]" (2019), http://arxiv.org/abs/1908.08835.

112 E. Chong, C. Han, F.C. Park, *Deep learning networks for stock market analysis and prediction: Methodology, data representations, and case studies*, „Expert Systems with Applications" t. 83 (2017), DOI: 10.1016/j.eswa.2017.04.030.

113 X. Jin, Z. Li, K. Liu, D. Zou, X. Li, X. Zhu, Z. Zhou, Q. Sun, Q. Liu, *Focusing on Persons: Colorizing Old Images Learning from Modern Historical Movies*, „arXiv:2108.06515 [cs]" (2021), http://arxiv.org/abs/2108.06515.

114 S. Ghose, J.J. Prevost, *AutoFoley: Artificial Synthesis of Synchronized Sound Tracks for Silent Videos with Deep Learning*, „IEEE Transactions on Multimedia" t. 23 (2021), DOI: 10.1109/TMM.2020.3005033.

115 K. Kamnitsas, C. Ledig, V.F.J. Newcombe, J.P. Simpson, A.D. Kane, D.K. Menon, D. Rueckert, B. Glocker, *Efficient Multi-Scale 3D CNN with Fully Connected CRF for Accurate Brain Lesion Segmentation*, „Medical Image Analysis" t. 36 (2017), DOI: 10.1016/j.media.2016.10.004.

116 J. Jumper, R. Evans, A. Pritzel, T. Green, M. Figurnov, O. Ronneberger, K. Tunyasuvunakool, R. Bates, A. Žídek, A. Potapenko, A. Bridgland, C. Meyer, S.A.A. Kohl, A.J. Ballard, A. Cowie, B. Romera-Paredes, S. Nikolov, R. Jain, J. Adler, T. Back, S. Petersen, D. Reiman, E. Clancy, M. Zielinski, M. Steinegger, M. Pacholska, T. Berghammer, S. Bodenstein, D. Silver, O. Vinyals, A.W. Senior, K. Kavukcuoglu, P. Kohli, D. Hassabis, *Highly accurate protein structure prediction with AlphaFold*, „Nature" t. 596 nr 7873 (2021), DOI: 10.1038/s41586-021-03819-2.

117 C. Xie, X.-X. Zhuang, Z. Niu, R. Ai, S. Lautrup, S. Zheng, Y. Jiang, R. Han, T.S. Gupta, S. Cao, M.J. Lagartos-Donate, C.-Z. Cai, L.-M. Xie, D. Caponio, W.-W. Wang, T. Schmauck-Medina, J. Zhang, H. Wang, G. Lou, X. Xiao, W. Zheng, K. Palikaras, G. Yang, K.A. Caldwell, G.A. Caldwell, H.-M. Shen, H. Nilsen, J.-H. Lu, E.F. Fang, *Amelioration of Alzheimer's disease pathology by mitophagy inducers identified via machine learning and a cross-species workflow*, „Nature Biomedical Engineering" t. 6 nr 1 (2022), DOI: 10.1038/s41551-021-00819-5.

118 Y. Alali, F. Harrou, Y. Sun, *A proficient approach to forecast COVID-19 spread via optimized dynamic machine learning models*, „Scientific Reports" t. 12 nr 1 (2022), DOI: 10.1038/s41598-022-06218-3.

119 J.A. Garrido Torres, V. Gharakhanyan, N. Artrith, T.H. Eegholm, A. Urban, *Augmenting zero-Kelvin quantum mechanics with machine learning for the prediction of chemical reactions at high temperatures*, „Nature Communications" t. 12 nr 1 (2021), DOI: 10.1038/s41467-021-27154-2.

leśnych[120], do optymalizacji metod redukcji dwutlenku węgla emitowanego do atmosfery[121], do identyfikacji czynników starzenia się ssaków[122], do pomocy osobom z niepełnosprawnościami (*e.g.* do pisania ręcznego poprzez wyobrażanie sobie tego procesu)[123], do zarządzania polem magnetycznym w reaktorach termojądrowych[124], do nawigowania balonami stratosferycznymi[125], w mechanice płynów[126], fizyce cząstek elementarnych[127], czy przewidywaniu zjawisk meteorologicznych[128], do efektywnego rozwiązywania kosztownych obliczeniowo problemów[129], oraz ogółem w matematyce i fizyce teoretycznej[130], *etc.*

Zakres zastosowań sieci neuronowych i uczenia maszynowego w kryminalistyce zarysować można na przykładzie Międzynarodowego Warsztatu Kryminalistyki Obliczeniowej (*International Workshop on Computational Forensics*, IWCF). Jest to

120 U. Dampage, L. Bandaranayake, R. Wanasinghe, K. Kottahachchi, B. Jayasanka, *Forest fire detection system using wireless sensor networks and machine learning*, „Scientific Reports" t. 12 nr 1 (2022), DOI: 10.1038/s41598-021-03882-9.

121 M. Zhong, K. Tran, Y. Min, C. Wang, Z. Wang, C.-T. Dinh, P. De Luna, Z. Yu, A.S. Rasouli, P. Brodersen, S. Sun, O. Voznyy, C.-S. Tan, M. Askerka, F. Che, M. Liu, A. Seifitokaldani, Y. Pang, S.-C. Lo, A. Ip, Z. Ulissi, E.H. Sargent, *Accelerated discovery of CO2 electrocatalysts using active machine learning*, „Nature" t. 581 nr 7807 (2020), DOI: 10.1038/s41586-020-2242-8.

122 S. Choi, D. Hill, L. Guo, R. Nicholas, D. Papadopoulos, M.F. Cordeiro, *Automated characterisation of microglia in ageing mice using image processing and supervised machine learning algorithms*, „Scientific Reports" t. 12 nr 1 (2022), DOI: 10.1038/s41598-022-05815-6.

123 F.R. Willett, D.T. Avansino, L.R. Hochberg, J.M. Henderson, K.V. Shenoy, *High-performance brain-to-text communication via handwriting*, „Nature" t. 593 nr 7858 (2021), DOI: 10.1038/s41586-021-03506-2.

124 J. Degrave, F. Felici, J. Buchli, M. Neunert, B. Tracey, F. Carpanese, T. Ewalds, R. Hafner, A. Abdolmaleki, D. de las Casas, C. Donner, L. Fritz, C. Galperti, A. Huber, J. Keeling, M. Tsimpoukelli, J. Kay, A. Merle, J.-M. Moret, S. Noury, F. Pesamosca, D. Pfau, O. Sauter, C. Sommariva, S. Coda, B. Duval, A. Fasoli, P. Kohli, K. Kavukcuoglu, D. Hassabis, M. Riedmiller, *Magnetic control of tokamak plasmas through deep reinforcement learning*, „Nature" t. 602 nr 7897 (2022), DOI: 10.1038/s41586-021-04301-9.

125 M.G. Bellemare, S. Candido, P.S. Castro, J. Gong, M.C. Machado, S. Moitra, S.S. Ponda, Z. Wang, *Autonomous navigation of stratospheric balloons using reinforcement learning*, „Nature" t. 588 nr 7836 (2020), DOI: 10.1038/s41586-020-2939-8.

126 C. Lagemann, K. Lagemann, S. Mukherjee, W. Schröder, *Deep recurrent optical flow learning for particle image velocimetry data*, „Nature Machine Intelligence" t. 3 nr 7 (2021), DOI: 10.1038/s42256-021-00369-0.

127 D. Guest, K. Cranmer, D. Whiteson, *Deep Learning and its Application to LHC Physics*, „Annual Review of Nuclear and Particle Science" t. 68 nr 1 (2018), DOI: 10.1146/annurev-nucl-101917-021019.

128 S. Ravuri, K. Lenc, M. Willson, D. Kangin, R. Lam, P. Mirowski, M. Fitzsimons, M. Athanassiadou, S. Kashem, S. Madge, R. Prudden, A. Mandhane, A. Clark, A. Brock, K. Simonyan, R. Hadsell, N. Robinson, E. Clancy, A. Arribas, S. Mohamed, *Skilful precipitation nowcasting using deep generative models of radar*, „Nature" t. 597 nr 7878 (2021), DOI: 10.1038/s41586-021-03854-z.

129 *Optimizing the synergy between physics and machine learning*, „Nature Machine Intelligence" t. 3 nr 11 (2021), DOI: 10.1038/s42256-021-00416-w.

130 M.R. Douglas, *Machine learning as a tool in theoretical science*, „Nature Reviews Physics" (2022), DOI: 10.1038/s42254-022-00431-9.

biennale, którego pierwsza edycja odbyła się w 2007 roku (Manchester)[131], druga w 2008 (Washington)[132], trzecia w 2009 (The Hague)[133], czwarta w 2010 (Tokyo)[134], piąta w 2012 (Tsukuba)[135], szósta w 2014 (Stockholm)[136], siódma w 2018 (Beijing)[137], a ostatnia edycja w 2021 roku (Milan)[138]. Opublikowano tam ogółem 74 artykuły. Na temat metod badania dokumentów ukazało się 30 publikacji, w tym 11 dotyczyło badań pismoznawczych, zaś kolejne 5 lingwistyki sądowej. Ogólnie pojętej biometrii (w tym daktyloskopii, odontologii, fonoskopii i traseologii – pomijając badania dokumentów) poświęcono 25 publikacji. Natomiast modelowaniu i wizualizacji danych kryminalistycznych, oraz obróbce ich fotografii cyfrowych poświęcono 12 artykułów. Na temat oprogramowania przeznaczonego do praktycznych zastosowań w pracy ekspertów i organów ścigania (*e.g.* celem wsparcia prowadzących czynność przesłuchania), oraz ewaluacji wyników badań uzyskiwanych bez wykorzystania metod obliczeniowych, poświęcono 10 publikacji.

Przegląd najnowszej literatury wskazuje na wciąż rosnące zainteresowanie zastosowaniami uczenia maszynowego w kryminalistyce, *e.g.* do określania płci lub wieku ludzi na podstawie ich uzębienia[139], śliny[140],

131 N. Zhang, A. Abraham (red.), *Proceedings of the Third International Symposium on Information Assurance and Security, IAS 2007, August 29-31, 2007, Manchester, United Kingdom*, IEEE Computer Society 2007, s. 383–446.

132 S.N. Srihari, K. Franke (red.), *Computational Forensics: Second International Workshop, IWCF 2008, Washington, DC, USA, August 7-8, 2008. Proceedings*, t. 5158, Berlin, Heidelberg 2008.

133 N. Zhang, A. Abraham (red.), *Proceedings of the Third International Symposium on Information Assurance and Security, IAS 2007, August 29-31, 2007, Manchester, United Kingdom*, IEEE Computer Society 2007.

134 H. Sako, K.Y. Franke, S. Saitoh (red.), *Computational Forensics: 4th International Workshop, IWCF 2010 Tokyo, Japan, November 11-12, 2010 Revised Selected Papers*, t. 6540, Berlin, Heidelberg 2011.

135 U. Garain, F. Shafait (red.), *Computational Forensics: 5th International Workshop, IWCF 2012 Tsukuba, Japan, November 11, 2012 and 6th International Workshop, IWCF 2014 Stockholm, Sweden, August 24, 2014 Revised Selected Papers*, t. 8915, Cham 2015.

136 Ibid.

137 Z. Zhang, D. Suter, Y. Tian, A. Branzan Albu, N. Sidère, H. Jair Escalante (red.), *Pattern Recognition and Information Forensics: ICPR 2018 International Workshops, CVAUI, IWCF, and MIPPSNA, Beijing, China, August 20-24, 2018, Revised Selected Papers*, t. 11188, Cham 2019.

138 A. Del Bimbo, R. Cucchiara, S. Sclaroff, G.M. Farinella, T. Mei, M. Bertini, H.J. Escalante, R. Vezzani (red.), *Pattern Recognition. ICPR International Workshops and Challenges: Virtual Event, January 10–15, 2021, Proceedings*, t. 1–8, Cham 2021.

139 M. Han, S. Du, Y. Ge, D. Zhang, Y. Chi, H. Long, J. Yang, Y. Yang, J. Xin, T. Chen, N. Zheng, Y. Guo, *With or without human interference for precise age estimation based on machine learning?*, „International Journal of Legal Medicine" (2022), DOI: 10.1007/s00414-022-02796-z, https://doi.org/10.1007/s00414-022-02796-z; K.C. Santosh, N. Pradeep, V. Goel, R. Ranjan, E. Pandey, P.K. Shukla, S.J. Nuagah, *Machine Learning Techniques for Human Age and Gender Identification Based on Teeth X-Ray Images*, „Journal of Healthcare Engineering" t. 2022 (2022), DOI: 10.1155/2022/8302674.

140 E. Buchan, L. Kelleher, M. Clancy, J.J. Stanley Rickard, P.G. Oppenheimer, *Spectroscopic molecular-fingerprint profiling of saliva*, „Analytica Chimica Acta" t. 1185 (2021), DOI:

kości[141] lub odcisków obuwia[142], do identyfikacji lub weryfikacji ludzi na podstawie ich głosu[143], do ustalania przyczyn[144] lub czasu[145] śmierci osób, do predykcji wyglądu osób na podstawie ich DNA[146], do analizy danych spektroskopowych i chromatograficznych (*e.g.* dla ustalenia przyczyn pożaru[147] lub identyfikacji substancji psychoaktywnych[148]), do analizy obrazów cyfrowych[149], w psychiatrii sądowej (*e.g*. dla oszacowania ryzyka związanego z funkcjonowaniem danej osoby w społeczeństwie)[150], celem analizy antropologicznej[151] (*e.g.* opisu uzębienia[152] lub uszkodzeń szkieletu[153]), podczas

10.1016/j.aca.2021.339074.

141 P. Intasuwan, P. Palee, A. Sinthubua, P. Mahakkanukrauh, *Comparison of sex determination using three methods applied to the greater sciatic notch of os coxae in a Thai population: Dry bone morphology, 2-dimensional photograph morphometry, and deep learning artificial neural network*, „Medicine, Science and the Law" (2022), DOI: 10.1177/00258024221079092.

142 M. Hassan, Y. Wang, D. Wang, D. Li, Y. Liang, Y. Zhou, D. Xu, *Deep learning analysis and age prediction from shoeprints*, „Forensic Science International" t. 327 (2021), DOI: 10.1016/j.forsciint.2021.110987.

143 K. V, S.P. S, *Hybrid machine learning classification scheme for speaker identification*, „Journal of Forensic Sciences" (2022), DOI: 10.1111/1556-4029.15006.

144 F.-Y. Zhang, L.-L. Wang, W.-W. Dong, M. Zhang, D. Tash, X.-J. Li, S.-K. Du, H.-M. Yuan, R. Zhao, D.-W. Guan, *A preliminary study on early postmortem submersion interval (PMSI) estimation and cause-of-death discrimination based on nontargeted metabolomics and machine learning algorithms*, „International Journal of Legal Medicine" (2022), DOI: 10.1007/s00414-022-02783-4, https://doi.org/10.1007/s00414-022-02783-4.

145 L. Hu, Y. Xing, P. Jiang, L. Gan, F. Zhao, W. Peng, W. Li, Y. Tong, S. Deng, *Predicting the postmortem interval using human intestinal microbiome data and random forest algorithm*, „Science & Justice" t. 61 nr 5 (2021), DOI: 10.1016/j.scijus.2021.06.006.

146 E. Pośpiech, P. Teisseyre, J. Mielniczuk, W. Branicki, *Predicting Physical Appearance from DNA Data—Towards Genomic Solutions*, „Genes" t. 13 nr 1 (2022), DOI: 10.3390/genes13010121.

147 C. Bogdal, R. Schellenberg, M. Lory, M. Bovens, O. Höpli, *Recognition of gasoline in fire debris using machine learning: Part II, application of a neural network*, „Forensic Science International" t. 332 (2022), DOI: 10.1016/j.forsciint.2022.111177.

148 J. Klingberg, B. Keen, A. Cawley, D. Pasin, S. Fu, *Developments in high-resolution mass spectrometric analyses of new psychoactive substances*, „Archives of Toxicology" (2022), DOI: 10.1007/s00204-022-03224-2, https://doi.org/10.1007/s00204-022-03224-2; S.Y. Lee, S.T. Lee, S. Suh, B.J. Ko, H.B. Oh, *Revealing Unknown Controlled Substances and New Psychoactive Substances Using High-Resolution LC-MS/MS Machine Learning Models and the Hybrid Similarity Search Algorithm*, „Journal of Analytical Toxicology" (2021), DOI: 10.1093/jat/bkab098.

149 P. Yang, *Dual-Domain Fusion Convolutional Neural Network for Contrast Enhancement Forensics*, „Entropy (Basel, Switzerland)" t. 23 nr 10 (2021), DOI: 10.3390/e23101318.

150 I. Kudeikina, M. Loseviča, N.O. Gutorova, *Legal and practical problems of use of artificial intelligence-based robots in forensic psychiatry*, „Wiadomosci Lekarskie (Warsaw, Poland: 1960)" t. 74 nr 11 cz 2 (2021).

151 E. Bermejo, K. Taniguchi, Y. Ogawa, R. Martos, A. Valsecchi, P. Mesejo, O. Ibáñez, K. Imaizumi, *Automatic landmark annotation in 3D surface scans of skulls: Methodological proposal and reliability study*, „Computer Methods and Programs in Biomedicine" t. 210 (2021), DOI: 10.1016/j.cmpb.2021.106380.

152 M. Estai, M. Tennant, D. Gebauer, A. Brostek, J. Vignarajan, M. Mehdizadeh, S. Saha, *Deep learning for automated detection and numbering of permanent teeth on panoramic images*, „Dento Maxillo Facial Radiology" t. 51 nr 2 (2022), DOI: 10.1259/dmfr.20210296.

153 N. Dempsey, R. Bassed, R. Amarasiri, S. Blau, *Exploring the use of machine learning for the assessment of skeletal fracture morphology and differentiation between impact mechanisms: A pilot study*, „Journal of Forensic Sciences" (2022), DOI: 10.1111/1556-4029.14996; V. Ibanez, S. Gunz, S. Erne, E.J. Rawdon, G. Ampanozi, S. Franckenberg, T. Sieberth, R. Affolter, L.C. Ebert, A. Dobay,

formułowania wniosków opinii patologicznych[154], do wykrywania mikroskopowych próbek nasienia[155], do predykcji pojazdu jakim sprawca dokonał potrącenia pieszego[156], do analizy przestrzelin dla określenia odległości z jakiej strzelano[157], celem wykrywania zmanipulowanych materiałów audiowizualnych (nawet po ich wielokrotnym udostępnieniu przez kolejne osoby)[158], a szczególnie do wykrywania materiałów audiowizualnych spreparowanych z wykorzystaniem uczenia maszynowego (tzw. fałszerstwa głębokie, *deepfakes*)[159], *etc*.

Pośród zastosowań uczenia maszynowego w badaniach pismoznawczych wyróżnić można przede wszystkim: i) rozpoznawanie treści[160]; ii) identyfikację wykonawców (stwierdzenie, która spośród danej grupy osób jest wykonawcą materiału kwestionowanego)[161]; iii) weryfikację wykonawców (stwierdzenie, czy dane dwa materiały pochodzą od tego samego, czy od różnych wykonawców)[162]; iv) wykrywanie

RiFNet: Automated rib fracture detection in postmortem computed tomography, „Forensic Science, Medicine, and Pathology" (2021), DOI: 10.1007/s12024-021-00431-8.

154 H.H. de Boer, J. Fronczek, C.E.H. Berger, M. Sjerps, *The logic of forensic pathology opinion*, „International Journal of Legal Medicine" (2022), DOI: 10.1007/s00414-021-02754-1, https://doi.org/10.1007/s00414-021-02754-1.

155 R. Golomingi, C. Haas, A. Dobay, S. Kottner, L. Ebert, *Sperm hunting on optical microscope slides for forensic analysis with deep convolutional networks - a feasibility study*, „Forensic Science International. Genetics" t. 56 (2022), DOI: 10.1016/j.fsigen.2021.102602.

156 M. Casali, D. Malchiodi, C. Spada, A.M. Zanaboni, R. Cotroneo, D. Furci, A. Sommariva, U. Genovese, A. Blandino, *A pilot study for investigating the feasibility of supervised machine learning approaches for the classification of pedestrians struck by vehicles*, „Journal of Forensic and Legal Medicine" t. 84 (2021), DOI: 10.1016/j.jflm.2021.102256.

157 P. Oura, A. Junno, J.-A. Junno, *Deep learning in forensic shotgun pattern interpretation – A proof-of-concept study*, „Legal Medicine" t. 53 (2021), DOI: 10.1016/j.legalmed.2021.101960.

158 F. Marcon, C. Pasquini, G. Boato, *Detection of Manipulated Face Videos over Social Networks: A Large-Scale Study*, „Journal of Imaging" t. 7 nr 10 (2021), DOI: 10.3390/jimaging7100193.

159 A. Ismail, M. Elpeltagy, M. Zaki, K.A. ElDahshan, *Deepfake video detection: YOLO-Face convolution recurrent approach*, „PeerJ Computer Science" t. 7 (2021), DOI: 10.7717/peerj-cs.730.

160 R. Ahmed, K. Dashtipour, M. Gogate, A. Raza, R. Zhang, K. Huang, A. Hawalah, A. Adeel, A. Hussain, *Offline Arabic Handwriting Recognition Using Deep Machine Learning: A Review of Recent Advances* [w:] J. Ren, A. Hussain, H. Zhao, K. Huang, J. Zheng, J. Cai, R. Chen, Y. Xiao (red.), *Advances in Brain Inspired Cognitive Systems*, Cham 2020.

161 Z.-Y. He, Q.-H. Chen, D.-F. Chen, A neural network expert system for Chinese handwriting-based writer identification [w:] Proceedings. International Conference on Machine Learning and Cybernetics, t. 4, 2002; S. Srihari, Y.-C. Shin, S. Lee, V. Govindaraju, S.-H. Cha, C.I. Tomai, B. Zhang, A. Shekhawat, D. Bartnik, W. Yang, S. Setlur, P. Kilinskas, F. Kunderman, X. Liu, Z. Shi, V. Ramanaprasad, Method and apparatus for analyzing and/or comparing handwritten and/or biometric samples. United States Patent US7580551B1, złożony 30 Czerwca 2003, przyznany 25 Sierpnia 2009, https://patents.google.com/patent/US7580551/en; O. Sudana, I.W. Gunaya, I.K.G. Darma Putra, Handwriting identification using deep convolutional neural network method, „TELKOMNIKA (Telecommunication Computing Electronics and Control)" t. 18 nr 4 (2020), DOI: 10.12928/telkomnika.v18i4.14864.

162 Beatrice Drott, Thomas Hassan-Reza, *On-line Handwritten Signature Verification using Machine Learning Techniques with a Deep Learning Approach* [w:] Lund 2015; H. Srinivasan, S.N. Srihari, M.J. Beal, *Machine Learning for Signature Verification* [w:] P. K. Kalra, S. Peleg (red.), *Computer Vision, Graphics and Image Processing*, Berlin, Heidelberg 2006.

fałszerstw pisma[163] lub podpisów[164] (nie musi to być tożsame ani z weryfikacją, ani z identyfikacją); v) wykrywanie manipulacji lub fałszerstw dokumentów[165].

Powyższe zastosowania podzielić można na badania obrazów offlinowych (*offline*)[166] i onlinowych (*online*)[167]. Pierwsze dotyczą materiałów sporządzonych odręcznie i w formie analogowej, które są następnie digitalizowane celem analizy za pomocą modeli uczenia maszynowego. Drugie dotyczą materiałów sporządzanych odręcznie i w formie cyfrowej, *i.e.* za pomocą specjalistycznych urządzeń elektronicznych, które zapisują nie tylko obraz pisma, ale wszystkie informacje na temat jego powstawania (*e.g.* nacisk i kąt elektronicznego narzędzia pisarskiego). Jednakże, ten podstawowy podział jest niedoskonały, bowiem nie wszystkie podpisy wykonane na urządzeniach elektronicznych zawierają informacje o dynamice ich powstania, a nie można uznać ich za offlinowe. Stąd, bardziej zasadnym jest podział na statyczne (*static*) i dynamiczne (*dynamic*) obrazy pisma, bo chociaż obrazy statyczne zawierają cechy informujące o dynamice powstawania pisma, to same w sobie nie są obrazami dynamicznymi (ruchomymi), ponieważ nie pozwalają odtworzyć nagrania podpisu.

163 P. Roy, S. Bag, *Detection of Handwritten Document Forgery by Analyzing Writers' Handwritings* [w:] B. Deka, P. Maji, S. Mitra, D. K. Bhattacharyya, P. K. Bora, S. K. Pal (red.), *Pattern Recognition and Machine Intelligence*, Cham 2019.

164 S.-H. Cha, C.C. Tappert, M. Gibbons, Y.-M. Chee, *Automatic Detection Of Handwriting Forgery Using A Fractal Number Estimate Of Wrinkliness*, „International Journal of Pattern Recognition and Artificial Intelligence" t. 18 nr 07 (2004), DOI: 10.1142/S0218001404003642; T.M. Ghanim, A.M. Nabil, *Offline Signature Verification and Forgery Detection Approach* [w:] *2018 13th International Conference on Computer Engineering and Systems (ICCES)*, Cairo, Egypt 2018; S.J. Gideon, A. Kandulna, A.A. Kujur, A. Diana, K. Raimond, *Handwritten Signature Forgery Detection using Convolutional Neural Networks*, „Procedia Computer Science" t. 143 (2018), DOI: 10.1016/j.procs.2018.10.336; S. Lai, L. Jin, Y. Zhu, Z. Li, L. Lin, *SynSig2Vec: Forgery-free Learning of Dynamic Signature Representations by Sigma Lognormal-based Synthesis*, „IEEE Transactions on Pattern Analysis and Machine Intelligence" (2021), DOI: 10.1109/TPAMI.2021.3087619.

165 M. Bibi, A. Hamid, M. Moetesum, I. Siddiqi, *Document Forgery Detection using Printer Source Identification—A Text-Independent Approach* [w:] *2019 International Conference on Document Analysis and Recognition Workshops (ICDARW)*, t. 8, 2019; G. Jaiswal, A. Sharma, S.K. Yadav, *Deep feature extraction for document forgery detection with convolutional autoencoders*, „Computers & Electrical Engineering" t. 99 (2022), DOI: 10.1016/j.compeleceng.2022.107770; S.-J. Ryu, H.-Y. Lee, I.-W. Cho, H.-K. Lee, *Document Forgery Detection with SVM Classifier and Image Quality Measures* [w:] Y.-M. R. Huang, C. Xu, K.-S. Cheng, J.-F. K. Yang, M. N. S. Swamy, S. Li, J.-W. Ding (red.), *Advances in Multimedia Information Processing - PCM 2008*, Berlin, Heidelberg 2008.

166 M.M. Hameed, R. Ahmad, M.L.M. Kiah, G. Murtaza, *Machine learning-based offline signature verification systems: A systematic review*, „Signal Processing: Image Communication" t. 93 (2021), DOI: 10.1016/j.image.2021.116139.

167 C.S. Vorugunti, V. Pulabaigari, R.K.S.S. Gorthi, P. Mukherjee, *OSVFuseNet: Online Signature Verification by feature fusion and depth-wise separable convolution based deep learning*, „Neurocomputing" t. 409 (2020), DOI: 10.1016/j.neucom.2020.05.072.

Ponadto, rozróżnić należy ze względu na dane uczące, że niektóre modele przetwarzają surowe lub preprocesowane obrazy pisma[168], inne zaś przetwarzają cechy pisma wyekstraktowane z obrazów za pomocą innych metod zautomatyzowanych[169] lub wyekstraktowane z obrazów przez ekspertów[170], inne jeszcze modele przetwarzają dane będące kombinacją powyższych[171].

2.3. Zastosowania sztucznych sieci neuronowych w antykryminalistyce. Jednym z przełomowych osiągnięć w dziedzinie uczenia maszynowego były generatywno-adwersaryjne sieci neuronowe (*generative-adversarial neural networks*, GAN) opracowane przez Goodfellow'a *et al.* w 2014 roku[172]. Przypomnieć należy, że metoda ta polega na uczeniu wespół dwóch sieci neuronowych, gdzie zadaniem pierwszej jest wytwarzanie danych sztucznych, a zadaniem drugiej jest odróżnianie danych sztucznych od prawdziwych. Uczenie takich sieci jest niezwykle trudne i niestabilne[173], bowiem bardzo łatwo dochodzi do sytuacji, w których jedna sieć dominuje nad drugą. Podczas gdy, efektywne rezultaty osiągane są tylko wtedy, gdy obydwie sieci uczą się od siebie poprzez wzajemną konkurencję, gdzie żadna z nich nie

168 S. Fiel, R. Sablatnig, *Writer Identification and Retrieval Using a Convolutional Neural Network* [w:] G. Azzopardi, N. Petkov (red.), *Computer Analysis of Images and Patterns*, Cham 2015; S.J. Gideon, A. Kandulna, A.A. Kujur, A. Diana, K. Raimond, *Handwritten Signature Forgery Detection using Convolutional Neural Networks*, „Procedia Computer Science" t. 143 (2018), DOI: 10.1016/j.procs.2018.10.336; O. Sudana, I.W. Gunaya, I.K.G. Darma Putra, *Handwriting identification using deep convolutional neural network method*, „TELKOMNIKA (Telecommunication Computing Electronics and Control)" t. 18 nr 4 (2020), DOI: 10.12928/telkomnika.v18i4.14864.

169 S.-H. Cha, C.C. Tappert, M. Gibbons, Y.-M. Chee, Automatic Detection Of Handwriting Forgery Using A Fractal Number Estimate Of Wrinkliness, „International Journal of Pattern Recognition and Artificial Intelligence" t. 18 nr 07 (2004), DOI: 10.1142/S0218001404003642; G. Jaiswal, A. Sharma, S.K. Yadav, Deep feature extraction for document forgery detection with convolutional autoencoders, „Computers & Electrical Engineering" t. 99 (2022), DOI: 10.1016/j.compeleceng.2022.107770; V. Kulesh, K. Schaffer, I. Sethi, M. Schwartz, Handwriting Quality Evaluation [w:] S. Singh, N. Murshed, W. Kropatsch (red.), Advances in Pattern Recognition — ICAPR 2001, t. 2013, Berlin, Heidelberg 2001; S. Srihari, Y.-C. Shin, S. Lee, V. Govindaraju, S.-H. Cha, C.I. Tomai, B. Zhang, A. Shekhawat, D. Bartnik, W. Yang, S. Setlur, P. Kilinskas, F. Kunderman, X. Liu, Z. Shi, V. Ramanaprasad, Method and apparatus for analyzing and/or comparing handwritten and/or biometric samples. United States Patent US7580551B1, złożony 30 Czerwca 2003, przyznany 25 Sierpnia 2009, https://patents.google.com/patent/US7580551/en.

170 H. Li, S. Fang, S. Mukhopadhyay, A.J. Saykin, L. Shen, Interactive Machine Learning by Visualization: A Small Data Solution, „Proceedings : IEEE International Conference on Big Data. IEEE International Conference on Big Data" t. 2018 (2018), DOI: 10.1109/BigData.2018.8621952.

171 C.S. Vorugunti, V. Pulabaigari, R.K.S.S. Gorthi, P. Mukherjee, *OSVFuseNet: Online Signature Verification by feature fusion and depth-wise separable convolution based deep learning*, „Neurocomputing" t. 409 (2020), DOI: 10.1016/j.neucom.2020.05.072.

172 I.J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, *Generative Adversarial Networks*, 10 czerwca 2014 r., http://arxiv.org/abs/1406.2661.

173 L. Weng, *From GAN to WGAN*, arXiv, 18 kwietnia 2019 r., http://arxiv.org/abs/1904.08994.

zdobywa dominującej przewagi. Metaforą takiej relacji mogą być dwaj szachiści, którzy konkurując ze sobą uczą się grać coraz lepiej, ale gdy jeden z nich zdobywa dominującą przewagę, to obydwaj przestają czynić jakiekolwiek postępy.

Rysunek 2.3.1. Fotografia (zatytułowana *Migrant Mother*, autorstwa Dorothea'y Lange, rok 1936) pokolorowana przez sieć generatywno-adwersaryjną.
Źródło: https://github.com/jantic/DeOldify, dostęp 14 luty 2023 r.

Metoda powyższa zyskała ogromną popularność i mnogie zastosowania, ale umożliwiła też fałszowanie danych na ogromną skalę. Pośród pozytywnych w praktyce zastosowań wyróżnić można *i.a.*: i) podnoszenie rozdzielczości[174], które znajduje bezpośrednie zastosowanie w telewizorach o jakości 8K (sygnał przesyłany jest w jakości 4K, a następnie sieć podnosi go do 8K)[175]; ii) kolorowanie fotografii wykonanych w skali szarości (rys. 2.3.1)[176]; iii) udźwiękawianie filmów[177]; iv) rekonstrukcje dzieł sztuki (rys. 2.3.2)[178]; v) oczyszczanie danych z szumu[179]; vi)

174 C. Ledig, L. Theis, F. Huszar, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, W. Shi, *Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network*, „arXiv:1609.04802 [cs, stat]" (2017), http://arxiv.org/abs/1609.04802.

175 *QLED 8K: Where AI Upscaling Meets Deep Learning* [na:] https://news.samsung.com/global/the-future-of-viewing-1-qled-8k-where-ai-upscaling-meets-deep-learning, dostęp 18 sierpnia 2022 r.

176 Q. Poterek, P.-A. Herrault, G. Skupinski, D. Sheeren, *Deep Learning for Automatic Colorization of Legacy Grayscale Aerial Photographs*, „IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing" t. 13 (2020), DOI: 10.1109/JSTARS.2020.2992082.

177 S. Ghose, J.J. Prevost, *AutoFoley: Artificial Synthesis of Synchronized Sound Tracks for Silent Videos with Deep Learning*, „IEEE Transactions on Multimedia" t. 23 (2021), DOI: 10.1109/TMM.2020.3005033.

178 I. Davis-Marks, *Lost Edges of Rembrandt's „Night Watch" Are Restored Using Artificial Intelligence* [na:] „Smithsonian Magazine", https://www.smithsonianmag.com/smart-news/lost-edges-rembrandts-night-watch-are-restored-using-artificial-intelligence-180978056/, dostęp 18 sierpnia 2022 r.

179 F. Mameli, M. Bertini, L. Galteri, A. Del Bimbo, *A NoGAN approach for image and video restoration and compression artifact removal* [w:] *2020 25th International Conference on Pattern Recognition*

tworzenie obrazów na podstawie szkicu[180] lub opisu (rys. 2.3.3)[181]; vii) tworzenie muzyki[182] i mowy[183] naśladujących dany styl lub autora; viii) translacja obrazów, *e.g.* przekształcanie zdjęć satelitarnych w mapy (rys. 2.3.4)[184].

Rysunek 2.3.2. Portret (zatytułowany Straż Nocna, Rembrandt van Rijn, rok 1642), którego brakujące brzegi zrekonstruowane zostały za pomocą sieci neuronowych (odkreślone białą linią).
Źródło: https://medium.com/geekculture/thanks-to-ai-the-night-watch-is-complete-after-300-years-a7ec61e6f859, dostęp 14 luty 2023 r.

(ICPR), Milan, Italy 2021.

180 T. Park, M.-Y. Liu, T.-C. Wang, J.-Y. Zhu, *Semantic Image Synthesis with Spatially-Adaptive Normalization*, arXiv, 5 listopada 2019 r., http://arxiv.org/abs/1903.07291.

181 A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, M. Chen, *Hierarchical Text-Conditional Image Generation with CLIP Latents*, arXiv, 12 kwietnia 2022 r., http://arxiv.org/abs/2204.06125.

182 P. Dhariwal, H. Jun, C. Payne, J.W. Kim, A. Radford, I. Sutskever, *Jukebox: A Generative Model for Music*, arXiv, 30 kwietnia 2020 r., http://arxiv.org/abs/2005.00341.

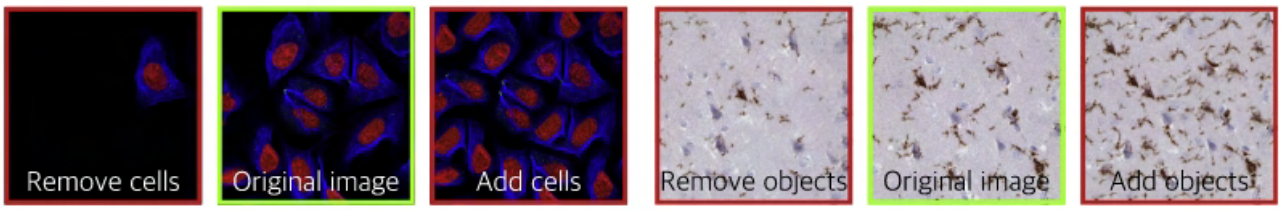
183 N. McGreevy, *Hear an A.I.-Generated Andy Warhol „Read" His Diary to You in New Documentary* [na:] „Smithsonian Magazine", https://www.smithsonianmag.com/smart-news/an-ai-generated-andy-warhol-reads-his-diary-to-you-in-new-documentary-180979658/, dostęp 18 sierpnia 2022 r.

184 P. Isola, J.-Y. Zhu, T. Zhou, A.A. Efros, *Image-to-Image Translation with Conditional Adversarial Networks*, 26 listopada 2018 r., http://arxiv.org/abs/1611.07004.

Rysunek 2.3.3. Obrazy wygenerowane przez dyfuzyjną sieć generatywno-adwersaryjną DALL·E 2 na podstawie danych tekstowych (pod obrazami), które opisywały co ma być na tych obrazach widoczne. Źródło: A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, M. Chen, *Hierarchical Text-Conditional Image Generation with CLIP Latents*, arXiv, 12 kwietnia 2022 r., http://arxiv.org/abs/2204.06125.

Rysunek 2.3.4. Zdjęcia satelitarne przekształcone w mapy przez sieć generatywno-adwersaryjną. Źródło: P. Isola, J.-Y. Zhu, T. Zhou, A.A. Efros, *Image-to-Image Translation with Conditional Adversarial Networks*, „arXiv:1611.07004 [cs]" (2018), http://arxiv.org/abs/1611.07004.

Przykłady powyższe świadczą też o możliwych zastosowaniach sieci generatywno-adwersaryjnych do generowania fałszywych danych na przykładzie danych prawdziwych, *i.a.:* i) do generowania mowy danych osób[185]; ii) do generowania materiałów audiowizualnych przedstawiających daną osobę i jej wypowiedź[186]; iii) do generowania audiowizualnych materiałów pornograficznych, gdzie wizerunek osoby nakładany jest na wizerunek aktora[187]; iv) do fałszowania transakcji i sprawozdań finansowych w sposób potencjalnie niewykrywalny dla audytorów i automatycznych systemów audytorskich[188]; v) a nawet do fałszowania badań naukowych (rys. 2.3.5)[189]. Dane fałszowane w ten sposób, ze względu na zastosowanie głębokich sieci neuronowych, przyjęło się określać mianem fałszerstw głębokich (*deepfakes*).



Rysunek 2.3.5. Przykład modyfikacji zdjęć mikroskopowych, przeprowadzonej za pomocą sieci generatywno-adwersaryjnych (oryginalne zdjęcia oznaczono zieloną ramką, zdjęcia na których sieć dokonała usunięcia lub dodania obiektów oznaczono ramką czerwoną).
Źródło: J. Gu, X. Wang, C. Li, J. Zhao, W. Fu, G. Liang, J. Qiu, *AI-enabled image fraud in scientific publications*, „Patterns" t. 3 nr 7 (2022), DOI: 10.1016/j.patter.2022.100511.

Problematyka wykrywania fałszerstw popełnionych za pomocą sieci generatywno-adwersaryjnych jest bardzo szeroka, wyróżnić w niej można jednak kilka głównych metod: i) powierzchowna inspekcja wzrokowa, która pozwala odrzucać gorszej jakości fałszerstwa w dużych liczbach, ale jest zawodna wobec bardziej

185 N. McGreevy, *Hear an A.I.-Generated Andy Warhol „Read" His Diary to You in New Documentary* [na:] „Smithsonian Magazine", https://www.smithsonianmag.com/smart-news/an-ai-generated-andy-warhol-reads-his-diary-to-you-in-new-documentary-180979658/, dostęp 18 sierpnia 2022 r.
186 S. Suwajanakorn, S.M. Seitz, I. Kemelmacher-Shlizerman, *Synthesizing Obama: learning lip sync from audio*, „ACM Transactions on Graphics" t. 36 nr 4 (2017), DOI: 10.1145/3072959.3073640.
187 C. Öhman, *Introducing the pervert's dilemma: a contribution to the critique of Deepfake Pornography*, „Ethics and Information Technology" t. 22 nr 2 (2020), DOI: 10.1007/s10676-019-09522-1.
188 M. Schreyer, T. Sattarov, B. Reimer, D. Borth, *Adversarial Learning of Deepfakes in Accounting*, arXiv, 9 października 2019 r., http://arxiv.org/abs/1910.03810.
189 J. Gu, X. Wang, C. Li, J. Zhao, W. Fu, G. Liang, J. Qiu, *AI-enabled image fraud in scientific publications*, „Patterns" t. 3 nr 7 (2022), DOI: 10.1016/j.patter.2022.100511.

wyrafinowanych fabrykacji[190]; ii) ocena danych za pomocą sieci neuronowych[191], które są wysoce skuteczne w wykrywaniu artefaktów świadczących o fałszerstwach głębokich (GAN *fingerprints*)[192], gdzie artefakty te posłużyć mogą do zidentyfikowania konkretnej sieci fałszującej[193] (w konsekwencji, sieci służące do wykrywania fałszerstw gorzej generalizują i bardziej się przeuczają)[194]; iii) połączenie metody pierwszej i drugiej, które umożliwiać może osiąganie lepszych wyników niż zastosowanie tych metod osobno, ponieważ są podobnie trafne ale zawodzą w różnych sytuacjach[195]; iv) manualne lub automatyczne wykrywanie i porównywanie arbitralnie określonych cech, takich jak częstotliwość mrugania oczu[196], manieryzmy[197], widmo sygnału (rys. 2.3.6)[198], niekonsekwentne pozycje twarzy na głowie[199], desynchronizacja dźwięku i obrazu[200]

190 M. Groh, Z. Epstein, C. Firestone, R. Picard, *Deepfake detection by human crowds, machines, and machine-informed crowds*, „Proceedings of the National Academy of Sciences" t. 119 nr 1 (2022), DOI: 10.1073/pnas.2110013119; J. Gu, X. Wang, C. Li, J. Zhao, W. Fu, G. Liang, J. Qiu, *AI-enabled image fraud in scientific publications*, „Patterns" t. 3 nr 7 (2022), DOI: 10.1016/j.patter.2022.100511.

191 D. Afchar, V. Nozick, J. Yamagishi, I. Echizen, *MesoNet: a Compact Facial Video Forgery Detection Network* [w:] *2018 IEEE International Workshop on Information Forensics and Security (WIFS)*, 2018; D. Güera, E.J. Delp, *Deepfake Video Detection Using Recurrent Neural Networks* [w:] *2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, 2018; H. Zhao, W. Zhou, D. Chen, T. Wei, W. Zhang, N. Yu, *Multi-Attentional Deepfake Detection* [w:] 2021.

192 L. Deng, H. Suo, D. Li, *Deepfake Video Detection Based on EfficientNet-V2 Network*, „Computational Intelligence and Neuroscience" t. 2022 (2022), DOI: 10.1155/2022/3441549.

193 N. Yu, L. Davis, M. Fritz, *Attributing Fake Images to GANs: Learning and Analyzing GAN Fingerprints*, 16 sierpnia 2019 r., http://arxiv.org/abs/1811.08180.

194 A.R. Javed, Z. Jalil, W. Zehra, T.R. Gadekallu, D.Y. Suh, Md.J. Piran, *A comprehensive survey on digital video forensics: Taxonomy, challenges, and future directions*, „Engineering Applications of Artificial Intelligence" t. 106 (2021), DOI: 10.1016/j.engappai.2021.104456; Y. Mirsky, W. Lee, *The Creation and Detection of Deepfakes: A Survey*, 31 stycznia 2022 r., http://arxiv.org/abs/2004.11138; Z. Wang, Y. Guo, W. Zuo, *Deepfake Forensics via an Adversarial Game*, „IEEE Transactions on Image Processing" t. 31 (2022), DOI: 10.1109/TIP.2022.3172845; P. Yu, Z. Xia, J. Fei, Y. Lu, *A Survey on Deepfake Video Detection*, „IET Biometrics" t. 10 nr 6 (2021), DOI: 10.1049/bme2.12031.

195 M. Groh, Z. Epstein, C. Firestone, R. Picard, *Deepfake detection by human crowds, machines, and machine-informed crowds*, „Proceedings of the National Academy of Sciences" t. 119 nr 1 (2022), DOI: 10.1073/pnas.2110013119.

196 T. Jung, S. Kim, K. Kim, *DeepVision: Deepfakes Detection Using Human Eye Blinking Pattern*, „IEEE Access" t. 8 (2020), DOI: 10.1109/ACCESS.2020.2988660; Y. Li, M.-C. Chang, S. Lyu, *In Ictu Oculi: Exposing AI Created Fake Videos by Detecting Eye Blinking* [w:] 2018.

197 S. Agarwal, H. Farid, Y. Gu, M. He, K. Nagano, H. Li, *Protecting World Leaders Against Deep Fakes* [w:] *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2019.

198 J. Frank, T. Eisenhofer, L. Schönherr, A. Fischer, D. Kolossa, T. Holz, *Leveraging Frequency Analysis for Deep Fake Image Recognition*, arXiv, 26 czerwca 2020 r., http://arxiv.org/abs/2003.08685.

199 K. Lutz, R. Bassett, *DeepFake Detection with Inconsistent Head Poses: Reproducibility and Analysis*, „ArXiv" (2021).

200 K. Chugh, P. Gupta, A. Dhall, R. Subramanian, *Not made for each other- Audio-Visual Dissonance-based Deepfake Detection and Localization* [w:] *Proceedings of the 28th ACM International Conference on Multimedia*, New York, NY, USA 2020.

zniekształcenia cech twarzy[201], tekstury skóry[202] i rytmu pracy serca (rys. 2.3.7)[203], *etc.*, oraz zespołów tych cech[204].

Rysunek 2.3.6. Przykład widma częstotliwościowego obliczonego za pomocą dyskretnej transformaty kosinusowej (*Discrete Cosine Transform*, DCT) dla obrazu prawdziwego z bazy FFHQ (po lewej) i wytworzonego przez sieć fałszującą (po prawej), która uczona była na obrazach z bazy FFHQ. Źródło: J. Frank, T. Eisenhofer, L. Schönherr, A. Fischer, D. Kolossa, T. Holz, *Leveraging Frequency Analysis for Deep Fake Image Recognition*, arXiv, 26 czerwca 2020 r., http://arxiv.org/abs/2003.08685.

Warto tutaj nadmienić, że wspominany rytm pracy serca mierzony jest tylko w przypadku nagrań wideo, na podstawie minimalnych zmian koloru skóry twarzy powodowanych cyklicznym krążeniem krwi (rys. 2.3.7), a wykrywany jest za pomocą metod zdalnej fotopletyzmografii (*remote photoplethysmography*, rPPG)[205]. Metoda ta znajduje zastosowanie w kryminalistyce także do sprawdzania czy osoby widoczne na nagraniach wideo są żywe. Na przykład, aby chronić algorytmy rozpoznawania twarzy przed próbami ukrycia tożsamości (*presentation attacks*)[206], choćby przed osobami poruszającymi się w maskach.

201 Y. Li, S. Lyu, *Exposing DeepFake Videos By Detecting Face Warping Artifacts*, „CVPR Workshops" (2019).

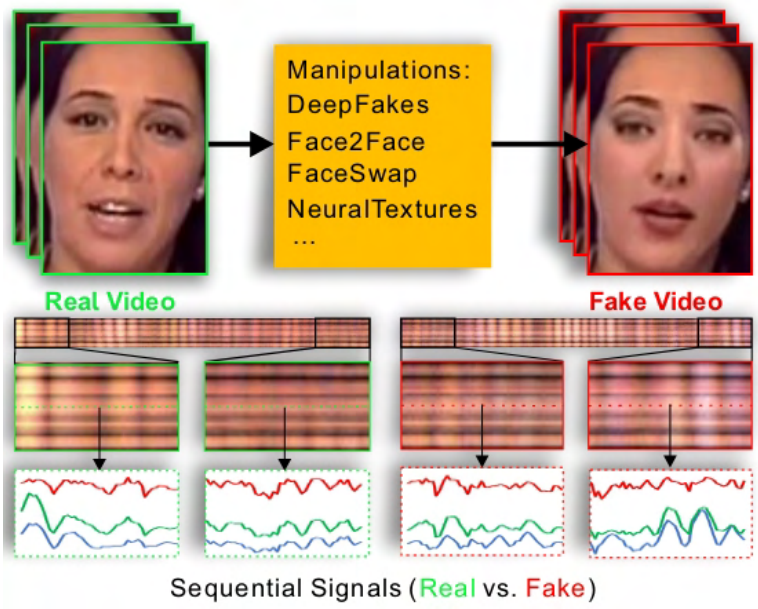
202 Z. Liu, X. Qi, P.H.S. Torr, *Global Texture Enhancement for Fake Face Detection in the Wild* [w:] *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Seattle, WA, USA 2020.

203 U.A. Ciftci, İ. Demir, L. Yin, *How Do the Hearts of Deep Fakes Beat? Deep Fake Source Detection via Interpreting Residuals with Biological Signals* [w:] *2020 IEEE International Joint Conference on Biometrics (IJCB)*, Houston, TX, USA 2020; H. Qi, Q. Guo, F. Juefei-Xu, X. Xie, L. Ma, W. Feng, Y. Liu, J. Zhao, *DeepRhythm: Exposing DeepFakes with Attentional Visual Heartbeat Rhythms*, „arXiv:2006.07634 [cs]" (2020), http://arxiv.org/abs/2006.07634.

204 M.F. Hashmi, B.K.K. Ashish, A.G. Keskar, N.D. Bokde, J.H. Yoon, Z.W. Geem, *An Exploratory Analysis on Visual Counterfeits Using Conv-LSTM Hybrid Architecture*, „IEEE Access" t. 8 (2020), DOI: 10.1109/ACCESS.2020.2998330.

205 A. Dasari, S.K.A. Prakash, L.A. Jeni, C.S. Tucker, *Evaluation of biases in remote photoplethysmography methods*, „npj Digital Medicine" t. 4 nr 1 (2021), DOI: 10.1038/s41746-021-00462-z.

206 Z. Yu, Y. Qin, X. Li, C. Zhao, Z. Lei, G. Zhao, *Deep Learning for Face Anti-Spoofing: A Survey*, arXiv, 2 września 2022 r., http://arxiv.org/abs/2106.14948.



Rysunek 2.3.7. Przykład zapisu rytmu pracy serca zmierzonego metodą rPPG na prawdziwym (po lewej, zielona ramka) i sztucznym nagraniu wideo (po prawej, czerwona ramka).

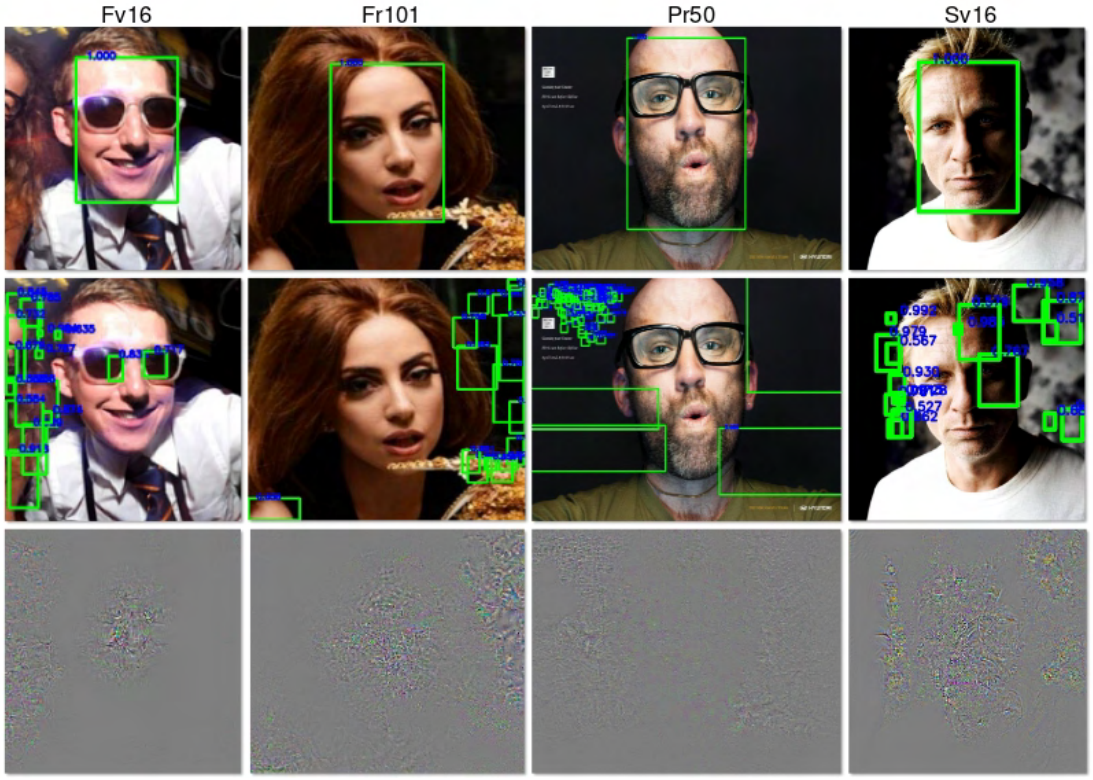
Źródło: H. Qi, Q. Guo, F. Juefei-Xu, X. Xie, L. Ma, W. Feng, Y. Liu, J. Zhao, *DeepRhythm: Exposing DeepFakes with Attentional Visual Heartbeat Rhythms*, „arXiv:2006.07634 [cs]" (2020), http://arxiv.org/abs/2006.07634.

Podstawowym zabezpieczeniem przed fałszerstwami popełnianymi za pomocą sieci generatywno-adwersaryjnych, po które sięgnąć może odbiorca danych, jest żądanie aby dane przedstawiane były w wysokiej rozdzielczości. Ponieważ, sieci generatywno-adwersaryjne radzą sobie z takimi danymi znacznie gorzej. W rezultacie, odbiorca albo danych takich nie otrzyma, albo większe będzie prawdopodobieństwo wykrycia fałszerstwa. Podstawowe metody zabezpieczania, po które sięgnąć może nadawca danych, polegają na nanoszeniu cyfrowych znaków wodnych lub podpisów elektronicznych[207]. Natomiast, bardziej zaawansowane metody polegają na nanoszeniu niewidocznego dla ludzi szumu cyfrowego, który wprowadza sieci fałszujące w błąd, uniemożliwiając im skuteczne fałszerstwa głębokie (rys. 2.3.8)[208]. Metody te korzystają z sieci generatywno-adwersaryjnych, które uczą się oszukiwać sieci służące do

207 A.R. Javed, Z. Jalil, W. Zehra, T.R. Gadekallu, D.Y. Suh, Md.J. Piran, *A comprehensive survey on digital video forensics: Taxonomy, challenges, and future directions*, „Engineering Applications of Artificial Intelligence" t. 106 (2021), DOI: 10.1016/j.engappai.2021.104456.

208 Y. Li, X. Yang, B. Wu, S. Lyu, *Hiding Faces in Plain Sight: Disrupting AI Face Synthesis with Adversarial Perturbations*, arXiv, 21 czerwca 2019 r., http://arxiv.org/abs/1906.09288; S. Shan, E. Wenger, J. Zhang, H. Li, H. Zheng, B.Y. Zhao, *Fawkes: protecting privacy against unauthorized deep learning models* [w:] *Proceedings of the 29th USENIX Conference on Security Symposium*, USA 2020.

wykrywania określonych obiektów (*e.g.* twarzy), poprzez generowanie niezauważalnego szumu cyfrowego, który nanoszony jest następnie na dane. Stąd, sieć fałszująca, która miałaby zmodyfikować zdjęcie twarzy chronione takim szumem adwersaryjnym, nie będzie zdolna tej twarzy rozpoznać, a więc i jej zmodyfikować.



Rysunek 2.3.8. Przykład ochrony danych poprzez nanoszenie szumu adwersaryjnego. W pierwszym rzędzie znajdują się fotografie niechronione, na których kolejne sieci neuronowe dokonały rozpoznania twarzy (sieci oznaczono jako Fv16, Fr101, Pr50, Sv16). W drugim rzędzie znajdują się fotografie chronione przez szum adwersaryjny, na których sieciom neuronowym nie udało się już żadnych twarzy trafnie zidentyfikować. W trzecim rzędzie znajduje się szum adwersaryjny, który nanoszono na zdjęcia z rzędu drugiego (trzydziestokrotnie wzmocniony, aby był widoczny gołym okiem).
Źródło: Y. Li, X. Yang, B. Wu, S. Lyu, *Hiding Faces in Plain Sight: Disrupting AI Face Synthesis with Adversarial Perturbations*, arXiv, 21 czerwca 2019 r., http://arxiv.org/abs/1906.09288

Ostatnie publikacje wskazują, że w wyścigu pomiędzy sieciami neuronowymi do fałszowania danych, a sieciami neuronowymi do wykrywania ich fałszerstw, te pierwsze zwiększają, niestety, swoje prowadzenie[209]. Ponadto, sieci neuronowe służące do wykrywania fałszerstw mogą być wykorzystywane jako subsydiarne dyskryminatory

209 Y. Li, X. Yang, P. Sun, H. Qi, S. Lyu, *Celeb-DF: A Large-scale Challenging Dataset for DeepFake Forensics*, arXiv, 16 marca 2020 r., http://arxiv.org/abs/1909.12962; Y. Mirsky, W. Lee, *The Creation and Detection of Deepfakes: A Survey*, 31 stycznia 2022 r., http://arxiv.org/abs/2004.11138.

podczas uczenia sieci fałszujących i przyczyniać się do podnoszenia jakości ich fałszerstw. Ponadto, celem ukrywania fałszerstw przed sieciami neuronowymi przeznaczonymi do ich wykrywania, nanosić można na fałszerstwa szum adwersaryjny[210]. Otwartym problemem pozostaje kwestia wykrywania fałszerstw głębokich w przypadkach, kiedy zostały one przekształcone do postaci analogowej (*e.g.* wywołane fałszywe fotografie)[211].

Podobnie, otwartą kwestią pozostaje wpływ szumu adwersaryjnego – stosowanego do ochrony danych – na trafność kryminalistycznych sieci neuronowych stosowanych do analizy tych danych. Na przykład, czy sieć neuronowa stosowana do badań daktyloskopijnych będzie skuteczna wobec próbek zabezpieczonych szumem adwersaryjnym? Przypuszczać również można, że nie tylko fałszerstwa głębokie, ale też dane prawdziwe zabezpieczać można przed kryminalistycznymi sieciami neuronowymi za pomocą szumu adwersaryjnego, który wprowadzać ma je w błąd. Także otwartą kwestią pozostaje problem zatruwania prawdziwych danych poprzez przetwarzanie ich przez identycznościowe sieci generatywno-adwersaryjne (*autoencoders*)[212], *i.e.* takie, gdzie jedna sieć dokonuje zakodowania danych, a druga ich wiernego odtworzenia, obarczając odtwarzane dane artefaktami mogącymi sugerować ich fałszerstwo.

210 N. Carlini, H. Farid, *Evading Deepfake-Image Detectors with White- and Black-Box Attacks*, arXiv, 1 kwietnia 2020 r., http://arxiv.org/abs/2004.00622; S. Hussain, P. Neekhara, M. Jere, F. Koushanfar, J. McAuley, *Adversarial Deepfakes: Evaluating Vulnerability of Deepfake Detectors to Adversarial Examples*, arXiv, 7 listopada 2020 r., http://arxiv.org/abs/2002.12749; N. Papernot, P. McDaniel, I. Goodfellow, *Transferability in Machine Learning: from Phenomena to Black-Box Attacks using Adversarial Samples*, arXiv, 23 maja 2016 r., http://arxiv.org/abs/1605.07277.

211 Y. Mirsky, W. Lee, *The Creation and Detection of Deepfakes: A Survey*, 31 stycznia 2022 r., http://arxiv.org/abs/2004.11138.

212 Ibid.

Rozdział 3. Aspekty prawne zastosowania sztucznych sieci neuronowych w kryminalistyce.

W naszym obecnym systemie, oraz w przeważającej większości innych systemów, kwestie wytwarzania, obrotu i zastosowania uczenia maszynowego nie są odpowiednio uregulowane[213]. Podzielając potrzebę uregulowania tego obszaru Komisja Europejska zaproponowała w 2021 roku projekt Aktu w sprawie sztucznej inteligencji (*Artificial Intelligence Act*)[214]. Na podstawie postępów prac legislacyjnych[215], przypuszczać można, że Akt przyjęty zostanie w 2023 roku.

W motywach projektu przeczytać można, że: „Celem niniejszego rozporządzenia jest poprawa funkcjonowania rynku wewnętrznego poprzez ustanowienie jednolitych ram prawnych, w szczególności w zakresie rozwoju, wprowadzania do obrotu i wykorzystywania sztucznej inteligencji zgodnie z wartościami Unii. [...] Sztuczna inteligencja to szybko rozwijająca się grupa technologii, które mogą przyczyniać się do wielu różnych korzyści społeczno-ekonomicznych we wszystkich gałęziach przemysłu i obszarach działalności społecznej [...] Jednocześnie sztuczna inteligencja może być źródłem ryzyka i szkody dla interesu publicznego i przywilejów [*i.e.* praw] chronionych prawem Unii, w zależności od okoliczności dotyczących jej konkretnego zastosowania i wykorzystania. Szkody te mogą być materialne lub niematerialne”[216]. Pośród szczegółowych celów Aktu wyróżnić można: i) zapewnienie aby systemy sztucznej inteligencji były bezpieczne w użyciu i zgodne z podstawowymi prawami oraz wartościami unijnymi; ii) zapewnienie pewności prawa, poprzez wprowadzenie jednolitych regulacji w skali Unii Europejskiej, aby ułatwić inwestycje i innowacje w dziedzinie sztucznej inteligencji; iii) ułatwienie rozwoju jednolitego rynku, który zapewni zgodne z prawem, bezpieczne i wiarygodne zastosowania sztucznej inteligencji, oraz zapobieże fragmentacji rynku; iv) poprawę

213 D. Szostek, *Is the Traditional Method of Regulation (the Legislative Act) Sufficient to Regulate Artificial Intelligence, or Should It Also Be Regulated by an Algorithmic Code?*, „Białostockie Studia Prawnicze” t. 26 nr 3 (2021), DOI: 10.15290/bsp.2021.26.03.03.

214 Wniosek dotyczący rozporządzenia Parlamentu Europejskiego i Rady (UE) nr COM/2021/206 final z dnia 21 kwietnia 2021 r. ustanawiającego zharmonizowane przepisy dotyczące sztucznej inteligencji (akt w sprawie sztucznej inteligencji) i zmieniającego niektóre akty ustawodawcze unii.

215 Publications Office of the European Union, *Follow the steps of procedure 2021/0106/COD* [na:] https://eur-lex.europa.eu/legal-content/EN/HIS/?uri=celex:52021PC0206, dostęp 6 marca 2023 r.

216 Wniosek dotyczący rozporządzenia Parlamentu Europejskiego i Rady (UE) nr COM/2021/206 final z dnia 21 kwietnia 2021 r. ustanawiającego zharmonizowane przepisy dotyczące sztucznej inteligencji (akt w sprawie sztucznej inteligencji) i zmieniającego niektóre akty ustawodawcze unii, s. 21–29.

zarządzania i egzekwowania przepisów dotyczących praw podstawowych i wymogów bezpieczeństwa mających zastosowanie wobec sztucznej inteligencji.

Definicję sztucznej inteligencji znajdziemy w Załączniku I do projektu Aktu, który stanowi, że są to: „a) mechanizmy uczenia maszynowego, w tym uczenie nadzorowane, uczenie się maszyn bez nadzoru i uczenie przez wzmacnianie, z wykorzystaniem szerokiej gamy metod, w tym uczenia głębokiego; b) metody oparte na logice i wiedzy, w tym reprezentacja wiedzy, indukcyjne programowanie (logiczne), bazy wiedzy, silniki inferencyjne i dedukcyjne, rozumowanie (symboliczne) i systemy ekspertowe; c) podejścia statystyczne, estymacja bayesowska, metody wyszukiwania i optymalizacji”[217]. Jest to definicja bardzo szeroka, która obejmuje swoim zakresem wszystkie dotąd znane rodzaje sztucznych sieci neuronowych, ale uwzględnia też jako sztuczną inteligencję metody bardziej konserwatywne, *e.g.* sieci Bayesowskie.

Podstawowy mechanizm proponowanego Aktu uzależnia wymogi dotyczące systemów sztucznej inteligencji od ryzyka związanego z zastosowaniami tych systemów. Systemy sztucznej inteligencji wysokiego ryzyka wyróżnia się ze względu na obszary w których są stosowane. Obszary te określa Komisja Europejska w Załączniku III do Aktu (na podstawie art. 7 w związku z art. 73 projektu Aktu), biorąc pod uwagę ryzyko szkody dla zdrowia i bezpieczeństwa lub ryzyko niekorzystnego wpływu na prawa podstawowe. Niezależnie od Załącznika III, art. 6 projektu Aktu stanowi, że systemami wysokiego ryzyka są takie systemy sztucznej inteligencji, które spełniają obydwa następujące warunki: i) stanowią komponent bezpieczeństwa produktu lub same są produktem objętym unijnym prawodawstwem harmonizacyjnym wymienionym w Załączniku II; ii) na podstawie unijnego prawodawstwa harmonizacyjnego wymienionego w Załączniku II podlegają ocenie zgodności przeprowadzanej przez osobę trzecią w celu wprowadzenia do obrotu lub oddania do użytku. Jak wskazano w motywach projektu aktu: „Przy klasyfikowaniu systemu sztucznej inteligencji jako systemu wysokiego ryzyka zasadnicze znaczenie ma skala szkodliwego wpływu wywieranego przez system sztucznej inteligencji na prawa podstawowe chronione na mocy Karty. Do praw tych należą: prawo do godności człowieka, poszanowanie życia prywatnego i rodzinnego, ochrona danych osobowych, wolność wypowiedzi i

217 Załączniki do wniosku dotyczącego rozporządzenia Parlamentu Europejskiego i Rady (UE) nr COM/2021/206 final z dnia 21 kwietnia 2021 r. ustanawiającego zharmonizowane przepisy dotyczące sztucznej inteligencji (akt w sprawie sztucznej inteligencji) i zmieniającego niektóre akty ustawodawcze unii, s. 1.

informacji, wolność zgromadzania się i stowarzyszania się oraz niedyskryminacja, ochrona konsumentów, prawa pracownicze, prawa osób niepełnosprawnych, prawo do skutecznego środka prawnego i dostępu do bezstronnego sądu, prawo do obrony i domniemania niewinności, prawo do dobrej administracji [...] dzieciom przysługują szczególne prawa zapisane w art. 24 Karty praw podstawowych UE oraz w Konwencji o prawach dziecka [...] które wymagają uwzględnienia słabości dzieci oraz zapewnienia im takiej ochrony i opieki, jaka jest konieczna dla ich dobra. Podstawowe prawo do wysokiego poziomu ochrony środowiska zapisane w Karcie i wdrażane w strategiach politycznych Unii również należy uwzględnić w ocenie powagi szkody, jaką może spowodować system sztucznej inteligencji, w tym w odniesieniu do zdrowia i bezpieczeństwa osób”[218].

W istotnym dla niniejszej pracy zakresie, Załącznik III stanowi, że systemami sztucznej inteligencji wysokiego ryzyka są systemy stosowane w obszarach ścigania przestępczości i sprawowania wymiaru sprawiedliwości. W obszarze ścigania przestępczości będą to systemy przeznaczone do wykorzystania przez organy ścigania: i) do oszacowania ryzyka popełnienia lub ponownego popełnienia przestępstwa przez osobę fizyczną, oraz do oszacowania ryzyka na jakie narażone są potencjalne ofiary przestępstw; ii) w celu wykrywania nieszczerości lub stanu emocjonalnego osoby fizycznej; iii) do wykrywania fałszerstw głębokich (*deepfakes*) iv) w celu oceny wiarygodności dowodów w toku śledztwa lub dochodzenia; v) w celu przewidywania wystąpienia lub ponownego wystąpienia rzeczywistego lub potencjalnego przestępstwa na podstawie profilowania osób fizycznych lub w celu oceny cech osobowości i charakterystyki lub wcześniejszego zachowania przestępnego osób fizycznych lub grup. W obszarze sprawowania wymiaru sprawiedliwości będą to systemy, które: „mają służyć organowi sądowemu pomocą w badaniu i interpretacji stanu faktycznego i przepisów prawa oraz w stosowaniu prawa do konkretnego stanu faktycznego”[219]. Jeżeli Załącznik III interpretowany będzie w taki sposób, że systemy sztucznej inteligencji stosowane przez ekspertów nie są systemami wysokiego ryzyka, nawet gdy ekspert

218 Wniosek dotyczący rozporządzenia Parlamentu Europejskiego i Rady (UE) nr COM/2021/206 final z dnia 21 kwietnia 2021 r. ustanawiającego zharmonizowane przepisy dotyczące sztucznej inteligencji (akt w sprawie sztucznej inteligencji) i zmieniającego niektóre akty ustawodawcze unii, s. 29.

219 Załączniki do wniosku dotyczącego rozporządzenia Parlamentu Europejskiego i Rady (UE) nr COM/2021/206 final z dnia 21 kwietnia 2021 r. ustanawiającego zharmonizowane przepisy dotyczące sztucznej inteligencji (akt w sprawie sztucznej inteligencji) i zmieniającego niektóre akty ustawodawcze unii, s. 6.

sporządza za ich pomocą opinię, to możliwym będzie przemycanie lub oddziaływanie na obszary ścigania przestępczości i sprawowania wymiaru sprawiedliwości za pomocą systemów, które nie będą spełniać wymogów stawianych w tych obszarach.

Ogólne wymogi dotyczące systemów sztucznej inteligencji wysokiego ryzyka streścić można następująco: i) ustanawia się system zarządzania ryzykiem, realizowany przez cały cykl życia systemu sztucznej inteligencji (art. 9); ii) sporządza się dokumentację techniczną w taki sposób, aby wykazać, że system sztucznej inteligencji wysokiego ryzyka spełnia wymogi określone w Akcie, oraz aby dostarczyć właściwym organom i jednostkom wszystkich informacji niezbędnych do oceny zgodności systemu z tymi wymogami (art. 11); iii) zapewnia się funkcję automatycznej rejestracji zdarzeń, która umożliwia monitorowanie działania systemu pod kątem występowania sytuacji, które mogą skutkować tym, że system będzie stwarzał ryzyko (art. 12); iv) system projektuje się tak, aby mogły go skutecznie nadzorować osoby fizyczne celem zapobiegania lub minimalizowania ryzyka dla zdrowia, bezpieczeństwa lub praw podstawowych (art. 14); v) system opracowuje się na podstawie zbiorów danych spełniających określone kryteria jakości (art. 10); vi) do systemu dołącza się instrukcję obsługi zawierającą zwięzłe, kompletne, poprawne i jasne informacje (art. 13); vii) system opracowuje się w sposób zapewniający wystarczającą przejrzystość jego działania, umożliwiającą użytkownikom interpretację wyników działania systemu i ich właściwe wykorzystanie (art. 13); viii) system opracowuje się w taki sposób, aby osiągał, z uwagi na swoje przeznaczenie, odpowiedni poziom trafności (dokładności), rzetelności (solidności) i cyberbezpieczeństwa oraz działał konsekwentnie pod tymi względami w całym cyklu swojego życia (art. 15). Poniżej omówione zostaną wybrane spośród wymienionych wymogów, które mają największe znaczenie dla tematu niniejszej rozprawy (wymienione w punktach v-viii, a wynikające z art. 10, 13 i 15).

Artykuł 10 projektu określa wymogi dotyczące danych i zarządzania danymi, stanowiąc w punkcie 2, że zbiory danych treningowych, walidacyjnych i testowych podlegają odpowiednim praktykom, które dotyczą w szczególności: i) projektowania danych; ii) gromadzenia danych; iii) preprocesowania danych; iv) formułowania założeń w odniesieniu do informacji, które maja być reprezentowane przez dane; v) uprzedniej oceny dostępności, ilości i przydatności zbiorów danych, które są potrzebne; vi) badania danych pod kątem ewentualnej tendencyjności (*bias*); vii) określenia

wszelkich możliwych luk w danych lub braków w danych oraz tego, w jaki sposób można zaradzić tym lukom i brakom. W perspektywie wyników badań, które omówione zostaną w rozdziale 6, przytoczyć należy punkt 3, który stanowi, że: „[zbiory danych] muszą być adekwatne, reprezentatywne, wolne od błędów i kompletne. Muszą się one charakteryzować odpowiednimi właściwościami statystycznymi, w tym, w stosownych przypadkach, w odniesieniu do osób lub grup osób, wobec których ma być wykorzystywany system sztucznej inteligencji wysokiego ryzyka. Te kryteria zbiorów danych mogą zostać spełnione na poziomie pojedynczych zbiorów danych lub ich kombinacji”[220]. Natomiast, punkt 4 stanowi, że: „[zbiory danych] muszą uwzględniać, w zakresie wymaganym z uwagi na ich przeznaczenie, cechy lub elementy, które są specyficzne dla określonego kontekstu geograficznego, behawioralnego lub funkcjonalnego lub okoliczności, w których ma być wykorzystywany system sztucznej inteligencji wysokiego ryzyka”[221].

Artykuł 13 projektu określa kwestie przejrzystości i udostępniania informacji użytkownikom. Najważniejsze z tych informacji określają: i) poziom trafności, rzetelności i cyberbezpieczeństwa systemu, względem których przetestowano i walidowano system; ii) wszelkie znane i dające się przewidzieć okoliczności, które mogą mieć wpływ na oczekiwany poziom trafności, rzetelności i cyberbezpieczeństwa; iii) wszelkie znane lub dające się przewidzieć okoliczności związane z wykorzystaniem systemu, które mogą powodować zagrożenia dla zdrowia i bezpieczeństwa lub praw podstawowych; iv) poziom trafności i rzetelności systemu w odniesieniu do osób lub grup osób, względem których system ma być wykorzystywany. Ponadto, art. 13 nakłada niezwykle istotny obowiązek, stanowiąc, że systemy wysokiego ryzyka: „[...] opracowuje się w sposób zapewniający wystarczającą przejrzystość ich działania, umożliwiającą użytkownikom interpretację wyników działania systemu i ich właściwe wykorzystanie”[222]. W konsekwencji, sztuczne sieci neuronowe nie będą mogły być wykorzystywane jako systemy sztucznej inteligencji wysokiego ryzyka, ze względu na swoją nieinterpretowalność. Na przykładzie badań przeprowadzonych w rozdziale 7, wykazane jednak zostanie, że można opracować sztuczną sieć neuronową o adekwatnym poziomie interpretowalności.

220 Ibid., s. 57.
221 Ibid.
222 Ibid., s. 59.

Artykuł 15 projektu odnosi się do trafności, rzetelności i cyberbezpieczeństwa. Wymaga on, aby systemy sztucznej inteligencji wysokiego ryzyka: i) osiągały odpowiedni poziom trafności, rzetelności i cyberbezpieczeństwa oraz działały konsekwentnie pod tymi względami w całym cyklu życia; ii) były odporne na błędy, usterki lub niespójności, które mogą wystąpić w systemie lub w środowisku, w którym działa system; iii) nie stawały się tendencyjne w wyniku wykorzystywania ich danych wyjściowych jako danych wejściowych w przyszłych krokach (*feedback loops*); iv) aby były odporne na nieupoważnione próby mające na celu zmianę ich zastosowania lub skuteczności działania, *e.g*. poprzez ataki polegające na manipulacji zbiorem danych treningowych (*data poisoning*) lub danych wejściowych (*adversarial attacks*), które mają na celu spowodowanie błędu w modelu. Regulacje te dotyczą problemów ewaluacji i antykryminalistyki, których przykłady poruszono rozdziałach 6 i 8.

Obowiązki odnoszące się do systemów sztucznej inteligencji wysokiego ryzyka, które stawiane są wobec ich użytkowników, są następujące: i) użytkować system zgodnie z dołączoną do niego instrukcją obsługi; ii) zapewniać adekwatność danych wejściowych w odniesieniu do przeznaczenia systemu; iii) monitorować działanie systemu sztucznej inteligencji wysokiego ryzyka w oparciu o instrukcję obsługi; iv) wdrażać wskazane przez dostawcę środki nadzoru ze strony człowieka; v) jeżeli użytkowanie systemu zgodne z instrukcją obsługi może doprowadzić do powstania ryzyka, poinformować dostawcę lub dystrybutora o swoich przypuszczeniach i zaprzestać dalszego użytkowania systemu; vi) zgłaszać dostawcy lub dystrybutorowi wszelkie stwierdzone przez siebie poważne incydenty lub wszelkie przypadki nieprawidłowego działania i zaprzestać dalszego użytkowania systemu.

Na podstawie przeprowadzonych badań (rozdział 6, 7 i 8), projekt aktu ocenić należy pozytywnie z punktu widzenia kryminalistyki, skoro proponowane rygory będą adekwatne do ryzyka, jakie stwarzać może zastosowanie systemów sztucznej inteligencji w kryminalistyce. Niemniej, obawiać się można potencjalnej luki prawnej, jeżeli obszar opiniowania przez ekspertów nie będzie uwzględniany jako obszar wysokiego ryzyka, umożliwiając opiniowanie za pomocą systemów sztucznej inteligencji, nawet gdy nie będą spełniać one wymogów właściwych dla obszarów pokrewnych (ściganie przestępczości i sprawowanie wymiaru sprawiedliwości).

Rozdział 4. Aspekty praktyczne zastosowania sztucznych sieci neuronowych w kryminalistyce.

Rozważając aspekty praktyczne zastosowania sztucznych sieci neuronowych w kryminalistyce, autor przyjął za punkt odniesienia kwestię tego, czy udział eksperta jest konieczny do sporządzenia ekspertyzy maszynowej[223]. Umiejscowienie ekspertyzy maszynowej w danym porządku prawnym (*i.e.* to czy jest ona dopuszczalna, a udział biegłego konieczny), stanowi odmienny problem. Eksperta zdefiniowano tutaj jako osobę, która posiada tzw. wiadomości specjalne, czyli informacje wykraczające poza powszechną wiedzę przeciętnego człowieka[224]. Ekspertyza maszynowa rozumiana jest tutaj przede wszystkim jako rozstrzygnięcie problemu dokonane przez sztuczną sieć neuronową, ale odnosi się także do innych maszyn, które rozstrzygać mogą problemy kryminalistyczne. Wyróżniono następujące czynniki, od których zależeć będzie konieczność udziału eksperta przy sporządzeniu ekspertyzy maszynowej: i) prostota obsługi maszyny, a więc, czy potrzeba wiadomości specjalnych do jej obsługi; ii) samowystarczalność maszyny, a więc, czy dane wejściowe wymagają opracowania, a wyjściowe interpretacji przez eksperta; iii) trafność maszyny, a więc, czy wyniki zastosowania maszyny są tak samo pewne jak opinie ekspertów; iv) rzetelność maszyny, a więc, czy w różnych warunkach osiąga tą samą trafność; v) podatność maszyny na fałszerstwo, gdzie wyniki zastosowania maszyny podatnej na fałszerstwo będą bardziej wiarygodne, gdy ekspertyzę przeprowadzi ekspert; vi) pierwotność maszyny, a więc, czy posiada ona wiadomości specjalne pochodzące od ekspertów, które podważają konieczność ich udziału; vii) sprawdzalność maszyny, a więc, czy w oparciu o rozstrzygnięcia maszyny ekspert podejmował będzie dalsze działania, których powodzenie świadczyć będzie o poprawności rozstrzygnięcia maszyny, lub dodatkowe działania, które umożliwiać będą ewaluację maszyny; viii) interpretowalność maszyny, gdzie ekspert powinien sprawdzić poprawność metod realizowanych przez maszynę interpretowalną, lub dokonać ewaluacji adekwatnej do poziomu nieinterpretowalności maszyny. Do oceny tego, czy zastosowanie danej maszyny wymaga udziału eksperta,

223 Tematyka ta została też częściowo przedstawiona w artykule: M. Marcinowski, *Czy ekspertyza maszynowa wymaga wiadomości specjalnych?* [w:] *Wokół Kryminalistyki; Nauka i praktyka; Księga pamiątkowa dedykowana Profesorowi Tadeuszowi Widle*, Toruń 2021, s. 397–412.

224 K. Konieczny, T. Widła, J. Widacki, *Kryminalistyka*, Warszawa 2016.

potrzebne będą na ogół wiadomości specjalne z dziedziny uczenia maszynowego i dziedziny problemu, który maszyna rozwiązuje.

Prostota obsługi maszyny. Podstawową przesłanką dla udziału eksperta jest to, czy daną maszyną jest się łatwo posłużyć. W przypadku maszyn, których obsługa jest niezwykle trudna i skomplikowana, wymaga zaawansowanej wiedzy lub doświadczenia, a więc gdy wiadomości specjalne są konieczne, to udział eksperta będzie nieodzowny. W przypadku maszyn, którymi może posłużyć się przeciętna osoba zapoznawszy się z instrukcją obsługi, udział eksperta nie będzie konieczny.

Samowystarczalność maszyny. W przypadku każdej maszyny zastanowić się należy czy jest ona samowystarczalna, czy może wymaga ona uprzednich lub następujących po jej zastosowaniu działań ze strony osoby posiadającej wiadomości specjalne. Działania takie polegać mogą na: i) pozyskaniu lub opracowaniu danych wejściowych przez osobę posiadającą wiadomości specjalne; ii) interpretacji lub opracowaniu danych wyjściowych przez osobę posiadającą wiadomości specjalne. Jest to problem odmienny od tego, czy maszyną jest się łatwo posłużyć, bowiem nawet najprostsza w obsłudze maszyna wymagać może, aby pozyskać i wprowadzić do niej dane wejściowe, których pozyskanie lub opracowanie wymagać może udziału eksperta. Podobnie, nawet bardzo prosta w obsłudze maszyna może udzielać odpowiedzi, które wymagać będą dalszej interpretacji przez osobę posiadającą wiadomości specjalne.

Na przykład, jeżeli rzecz dotyczy badań, gdzie konieczne jest pobranie materiału przez osobę posiadającą wiadomości specjalne, to nie-ekspert wykonać ekspertyzy maszynowej nie może. Wątpliwym będzie także pobranie materiału przez ekspertów, a wykonanie ekspertyzy maszynowej przez nie-eksperta. Ponieważ, pobierając dane wejściowe o których rozstrzygać będzie maszyna, należy mieć na względzie właściwości tej maszyny, gdyż będzie to miało bezpośredni wpływ na jej rozstrzygnięcia. Podobnie, zastosować należy maszynę w sposób adekwatny do ilości i jakości pozyskanych danych.

Na ogół, modele uczenia maszynowego i sieci neuronowe częściej wymagać będą przygotowania lub pozyskania dla nich odpowiednich danych, niż poddania ich wyników interpretacji lub opracowaniu. Pierwszym przypadkiem będą tutaj rezultaty

maszyn interpretowalnych lub rezultaty probabilistyczne, które zawsze wymagać będą sprawdzenia i wyłożenia przez osobę posiadającą wiadomości specjalne. Drugim przypadkiem będą tutaj rozstrzygnięcia wieloproblemowe lub wieloaspektowe, gdzie maszyna rozwiązuje wiele problemów jednocześnie lub rozstrzyga o wielu aspektach jednego problemu jednocześnie (wielozadaniowość, *multitasking*). Potrzeba będzie tutaj wiadomości specjalnych, na przykład dlatego, że rozstrzygnięcia modelu wielozadaniowego mogą być wewnętrznie sprzeczne, o czym rozstrzygnąć będzie musiał ekspert. Trzecim przypadkiem będą maszyny, których zadaniem jest ułatwienie pracy eksperta przez wstępne opracowanie dla niego danych, które ekspert sam następnie opracowuje.

Trafność maszyny. Istotą problemu jest tutaj to, czy trafność rozstrzygnięć maszyny jest porównywalna do trafności ekspertyz wydawanych przez ekspertów, oraz to, czy udział eksperta podnosić może trafność ekspertyzy maszynowej w porównaniu do trafności ekspertyz maszynowych sporządzanych przez nie-ekspertów. Jeżeli nawet maszyna nie jest wystarczająco trafna, aby nie-ekspert mógł za jej pomocą daną kwestię trafnie rozstrzygnąć, to nie wyklucza to zastosowania tej maszyny przez eksperta, który będzie mógł zapobiegać jej błędom lub sanować jej wyniki za pomocą swoich wiadomości specjalnych.

Rzetelność maszyny. Określenie trafności maszyny jest pozornie prostym przedsięwzięciem, które sprowadza się do obliczenia wskaźnika trafności. Jednakże, jak wykazano w rozdziale 6 i 7, ewaluacja maszyny jest niełatwym zadaniem, które z jednej strony wymaga wiadomości specjalnych, a z drugiej jest ograniczone poziomem interpretowalności i falsyfikowalności danej maszyny. Dlatego, szczególnie ważnym jest określenie poziomu rzetelności maszyny, *i.e.*: czy w różnych warunkach maszyna osiąga podobną trafność. W przypadku maszyn o niskim poziomie rzetelności, a zawsze w przypadku sztucznych sieci neuronowych, przed zastosowaniem maszyny należało będzie określić jej trafność ze względu na: i) właściwości źródeł i danych o których maszyna będzie rozstrzygać (*e.g.* patologie pisma ręcznego)[225]; ii) warunki techniczne w których dane zostały pozyskane i zdigitalizowane (*e.g.* szum pochodzący

225 M. Całkiewicz, *Kryminalistyczne badania patologicznego pisma ręcznego*, Warszawa 2009.

ze skanera w przypadku digitalizacji dokumentów). W obydwu tych przypadkach udział eksperta będzie nieodzowny.

Podatność maszyny na fałszerstwo. Maszyny nie wystarczy poddać miarodajnej ewaluacji (określającej jej trafność i rzetelność), należy też bowiem zapytać o podatność danej maszyny i jej wskazań na fałszerstwo. Gdyby można było relatywnie łatwo i skutecznie manipulować lub fabrykować wyniki zastosowania danej maszyny, to koniecznym będzie zasięgnięcie opinii eksperta dla sprawdzenia czy do takowego zdarzenia doszło (oraz czy mogło być ono rezultatem błędu). Dlatego też, ekspertyza za pomocą takiej maszyny powinna znajdować się w rękach eksperta, bowiem podnosi to jej wiarygodność. Tym bardziej, tam gdzie bardzo łatwo i skutecznie wpłynąć można na wyniki maszyny, a bardzo trudno to wykryć, *i.e.* w przypadku uczenia maszynowego, to: i) osoby posiadające interes w wyniku ekspertyzy maszynowej nie powinny jej przeprowadzać; ii) ekspertyza taka powinna zostać przeprowadzona przez eksperta, który ręczy etyką zawodową.

Pierwotność maszyny. Rozstrzygnąć należy w przypadku każdego modelu, czy sam nauczył się rozwiązywać problem lub realizuje obiektywne prawa rzeczywistości (maszyna pierwotna), czy też uosabia on doświadczenie i wiedzę jakiegoś eksperta lub ich grupy (maszyna wtórna). W przypadku maszyn wtórnych (*e.g.* niektóre modele Bayesowskie), ekspertyza maszynowa powstaje w oparciu o wiadomości specjalne przekazane maszynie przez ekspertów, stąd udział eksperta nie jest konieczny. W przypadku zaś maszyn pierwotnych (*e.g.* sztuczne sieci neuronowe), realizują one własne wiadomości specjalne, które same nabyły w procesie uczenia maszynowego, a które nie muszą pokrywać się z metodyką zasadną do rozwiązywania danego typu problemów, stąd ich użycie powinno przebiegać pod kontrolą eksperta.

Sprawdzalność maszyny. Pytanie tutaj stanowi, czy dana maszyna jest sprawdzalna na podstawie czynności podejmowanych w skutek jej rozstrzygnięć. Gdyby tak bowiem było, a czynności te wymagały wiadomości specjalnych i umożliwiały sprawdzenie prawidłowości tych rozstrzygnięć, to zastosowanie maszyny stanowić powinno element wstępny tych czynności, które podjęte powinny być przez

eksperta. Jest to problem zbliżony do problemu samowystarczalności maszyn, gdzie potrzebne są wiadomości specjalne celem przygotowania danych wejściowych lub zinterpretowania rozstrzygnięć maszyny. Niemniej, jest to problem odmienny, ponieważ nawet gdy maszyna jest samowystarczalna (*i.e.* przyjmuje surowe dane i wydaje zrozumiałe dla ogółu rozstrzygnięcia), to: i) czynności podejmowane w skutek jej zastosowania wymagać mogą wiadomości specjalnych i prowadzić do sprawdzenia jej rozstrzygnięć; ii) wiadomości specjalne umożliwiać mogą podjęcie dodatkowych czynności w oparciu o rozstrzygnięcia maszyny, pozwalając na sprawdzenie jej rozstrzygnięć. Oddzielić tutaj należy jednak takie czynności, które byłyby podejmowane na podstawie rozstrzygnięć maszyny, ale uniemożliwiałyby dalsze ich sprawdzenie (*e.g.* wyrażenie zgody na kremację zwłok zidentyfikowanych przez maszyną).

W pierwszym przykładzie: i) pobrano próbki od danej osoby; ii) maszyna rozstrzygnęła o ich identyczności z materiałem kwestionowanym. Można przyjąć w takiej sytuacji, że nie jest sprawdzalna, bo na podstawie jej rozstrzygnięcia ani nie musi się, ani nie może się podjąć żadnych szczególnych czynności, które pozwalałyby potwierdzić lub zaprzeczyć jej rozstrzygnięciu. Oczywiście, ekspert może przeprowadzić własne badania i sam dokonać rozstrzygnięcia, ale podważałoby to sens uprzedniego zastosowania maszyny.

W drugim przykładzie: i) krąg osób podejrzanych był bardzo liczebny; ii) przyśpieszenia procesu identyfikacji dokonano za pomocą maszyny wstępnej selekcji próbek porównawczych; iii) maszyna dokonała pozytywnej identyfikacji danej liczby osób; iv) na podstawie czego ekspert przeprowadził swoje badania. Rozstrzygnięcie takiej maszyny zostało więc sprawdzone przez osobę posiadającą wiadomości specjalne konieczne do przeprowadzenia ekspertyzy, którą podjęto w skutek rozstrzygnięcia maszyny. W takich przypadkach, ekspertyza maszynowa nie funkcjonuje samoistnie, ale stanowi wstępny etap czynności podejmowanych przez eksperta.

W trzecim przykładzie: i) pobrano materiał porównawczy, który był istotnie różny od materiału kwestionowanego pod względem ilości i jakości; ii) zastosowano maszynę. Rozstrzygnięcie takiej maszyny jest sprawdzalne w wyniku dodatkowych czynności, jakie mogłaby podjąć osoba posiadająca wiadomości specjalne. Po pierwsze, osoba taka mogłaby zebrać materiał porównawczy o ilości i jakości podobnej do materiału kwestionowanego, w celu wydania własnej ekspertyzy lub uzupełnienia

ekspertyzy maszynowej. Po drugie, osoba taka mogłaby przetestować model na materiale porównawczym, sprawdzając jego trafność względem próbek o ilości i jakości podobnej do materiału kwestionowanego. W takich przypadkach, kiedy istnieje możliwość sprawdzenia lub uzupełnienia ekspertyzy maszynowej przez osoby posiadające wiadomości specjalne, to należy tak uczynić.

Interpretowalność maszyny. Jeżeli procesy decyzyjne maszyny są interpretowalne (semantycznie sensowne, bo znane są ich nazwy), to rozstrzygnąć należy: i) jaką metodę realizuje maszyna? ii) czy jest to metoda naukowa? iii) czy metoda ta została w danym przypadku poprawnie zastosowana przez maszynę? Gdyby odpowiedzi na te pytania były negatywne, to maszyny takiej nie należy stosować. Odpowiedzi na te pytania są konieczne i wymagają udziału eksperta. Pytania i-ii rozstrzygnąć można w sposób generalny, ale odpowiedź na pytanie iii wymaga udzielenia w każdym przypadku zastosowania maszyny.

Jeżeli procesy decyzyjne maszyny są nieinterpretowalne, to sprawdzić należy: i) czy była miarodajnie uczona i ewaluowana? ii) czy maszynę można zastosować w danym przypadku, ze względu na dostępną ilość i jakość danych, oraz zdolności generalizacyjne maszyny? Sprawdzenie kwestii powyższych wymaga udziału eksperta.

Ostatecznie, dla celów dowodowych opinie ekspertów muszą być transparentne, jeżeli więc ekspert nie będzie mógł posłużyć się maszyną, bo jest nieinterpretowalna i jego wiadomości specjalne nie mogą temu zaradzić, to maszyny takiej nikt nie powinien wykorzystywać w celach dowodowych. Dla celów niedowodowych, *e.g.* badań przesiewowych, zastosowania nieinterpretowalnej maszyny nie należy wykluczać, ale konieczny będzie udział eksperta (jak uzasadniono w poprzednim akapicie).

Rozdział 5. Aspekty metodologiczne zastosowania sztucznych sieci neuronowych w kryminalistyce.

Rozważając problem interpretacji i ewaluacji sztucznych sieci neuronowych jako metod kryminalistycznych, należy zwrócić uwagę, że naukowe metody kryminalistyczne opierają się na generalizacjach empirycznych, które uzyskiwane są za pomocą badań prowadzonych zgodnie z metodologią naukową. Podczas gdy, sztuczne sieci neuronowe same dokonują generalizacji empirycznych na podstawie danych uczących. Ponieważ są zaś nieinterpretowalne, to ich generalizacje są nieznanymi zdaniami, nie można więc ujmować ich jako hipotez. Stąd ewaluacja sztucznych sieci neuronowych, a *de facto* ich generalizacji, jest niezwykle trudna.

Celem ewaluacji powinna być ocena tego, czy dana metoda nadaje się do zastosowania w kryminalistyce. Można przytoczyć tutaj wiele kanonów oceny metod kryminalistycznych, jednakże dla autora szczególnie użyteczne były trzy następujące: i) standardy Dauberta[226]; ii) pytania J. Widackiego[227]; oraz kryteria T. Widły[228].

Pierwszym przykładem są tzw. standardy Dauberta (*Daubert standards*), które ustanowił Sąd Najwyższy Stanów Zjednoczonych Ameryki (*Supreme Court of the United States*, SCOTUS) orzeczeniem w sprawie *Daubert v. Merrell Dow Pharmaceuticals Inc.* z 1993 roku. Orzeczenie to jest na ogół pozytywnie postrzegane przez międzynarodową społeczność kryminalistyczną i stanowi dla niej wspólny punkt odniesienia[229] (nierzadko stanowi ono też punkt odniesienia dla prawodawców w innych krajach[230]). Wytyczne określone przez SCOTUS, którymi powinien kierować się sąd badając dopuszczalność opinii ze względu na daną metodę zastosowaną przez eksperta, przybierają postać następujących pytań: i) czy dana metoda jest powszechnie akceptowana w swojej dziedzinie naukowej? ii) czy dana metoda poddana została recenzowanej publikacji? iii) czy dana metoda została przetestowana lub może zostać przetestowana? iv) czy ustalono dla danej metody wskaźnik błędu i błąd ten jest

226 United States Supreme Court, *Daubert v. Merrell Dow Pharmaceuticals Inc., 509 U.S. 579* [w:] F. D. Wagner (red.), *United States Reports Volume 509, Cases Adjudged in the Supreme Court At October Term 1992*, Washington 1997.

227 K. Konieczny, T. Widła, J. Widacki, *Kryminalistyka*, Warszawa 2016, s. 447–452.

228 T. Widła, *Metodyka Ekspertyzy* [w:] M. Kała, D. Wild, J. Wójcikiewicz (red.), *Ekspertyza sądowa: zagadnienia wybrane*, Warszawa 2017, s. 29–45.

229 T. Tomaszewski, *Dowód z opinii biegłego w procesie karnym*, Kraków 2000, s. 121–125.

230 House of Commons Science and Technology Committee, *Forensic Science on Trial, Seventh Report of Session 2004–05, HC 96–I.*, London 2005, s. 75–76.

akceptowalny? v) czy dana metoda opracowana została na potrzeby danego postępowania (tzw. *litigation-related*) lub z zamiarem spowodowania określonego rozstrzygnięcia sądu? Szczególnie istotną jest tutaj obserwacja zawarta w uzasadnieniu, iż: „Metodologia naukowa oparta jest obecnie na generowaniu hipotez i ich testowaniu, aby sprawdzić czy mogą zostać sfalsyfikowane; w rzeczy samej, ta metodologia odróżnia naukę od innych form ludzkiego poznania [...] Twierdzenia, które stanowią wyjaśnienia naukowe, muszą być poddawalne testom empirycznym [tłum. własne]”[231].

Kolejnym przykładem są pytania, które zdaniem J. Widackiego powinien zadawać organ procesowy oceniając metodą zastosowaną przez biegłego[232]. Są one następujące: i) czy oferent metody jest uznanym autorytetem w środowisku naukowym i jest afiliowany w renomowanym ośrodku naukowym? ii) czy metoda opublikowana została w literaturze naukowej, a jej opis uwzględniał trafność (dokładność), rzetelność i umożliwia reprodukowalność metody? iii) czy stosowano już tą metodę w krajach, gdzie poziom nauki i ochrona praw człowieka są na najwyższym poziomie?

Ostatnim przykładem są kryteria, którymi zdaniem T. Widły powinien kierować się biegły przy wyborze metody koniecznej do sporządzenia opinii[233]. Są one następujące: i) trafność metody; ii) rzetelność (niezawodność) metody; iii) zasada ponoszenia kosztów rzeczywiście niezbędnych; iv) popularność metody.

Jak już zostało zauważone, procesy decyzyjne sztucznych sieci neuronowych są nieinterpretowalne, stąd ich generalizacje mają charakter nieznanych zdań i nie mogą być hipotezami. Powyższe przykłady wskazują jednak rozwiązanie tego problemu, postulując, że metoda kryminalistyczna powinna być zarówno trafna i rzetelna (jest to też zgodne z projektem Aktu w sprawie sztucznej inteligencji). Rzetelność (*reliability* lub *robustness*) definiowana będzie w niniejszej rozprawie jako zdolność metody do utrzymywania trafności pomimo zmieniających się warunków jej zastosowania. Nierzetelność stwierdzana będzie zaś w sytuacjach, kiedy różnice w poziomach trafności, będące skutkiem zastosowania metody w różnych warunkach, będą statystycznie istotne. Autor zakłada tutaj, że stwierdzenie nierzetelności metody tożsame jest z podważeniem generalizacji empirycznych, na których się ona opiera. Ponieważ,

231 United States Supreme Court, *Daubert v. Merrell Dow Pharmaceuticals Inc., 509 U.S. 579* [w:] F. D. Wagner (red.), *United States Reports Volume 509, Cases Adjudged in the Supreme Court At October Term 1992*, Washington 1997.

232 K. Konieczny, T. Widła, J. Widacki, *Kryminalistyka*, Warszawa 2016, s. 447–452.

233 T. Widła, *Metodyka Ekspertyzy* [w:] M. Kała, D. Wild, J. Wójcikiewicz (red.), *Ekspertyza sądowa: zagadnienia wybrane*, Warszawa 2017, s. 29–45.

generalizacje empiryczne o naukowym charakterze, gwarantować powinny trafność metody niezależną od zmienności warunków jej zastosowania.

Zakładając, że ewaluacja sieci neuronowych polegać będzie na badaniu ich rzetelności, określić należy kiedy poszczególne testy statystyczne (kwestionujące rzetelność modelu w poszczególnych warunkach), przesądzać będą o wartości hipotezy ogólnej, stanowiącej, że model jest rzetelny. Wskazówką dla rozwiązania tego problemu jest uzasadnienie orzeczenia w sprawie Dauberta, gdzie SCOTUS odwołuje się do falsyfikacjonizmu K. Poppera[234]. Kryterium falsyfikowalności wymaga, aby tak formułować hipotezy, żeby możliwym było ich obalanie. Zdaniem K. Poppera pozwala to oddzielić naukę od nie-nauki. Ponieważ, na podstawie *modus tollendo tollens,* udana falsyfikacja pozwala odrzucić hipotezę definitywnie fałszywą, zaś nieudana falsyfikacja pozwala uznać ją za teorię, aż do czasu gdy zostanie sfalsyfikowana. W tym przypadku, generalizacje empiryczne, na których opiera się metoda kryminalistyczna, muszą być: i) ewaluowalne (*i.e.* falsyfikowalne i weryfikowalne); ii) oraz przybierać postać wyjaśnienia naukowego (*i.e*. zdań, a w efekcie hipotez), czyli być interpretowalne. Również z punktu widzenia falsyfikacjonizmu, problem ewaluacji sztucznych sieci neuronowych wynika z ich nieinterpretowalności. Skoro bowiem generalizacje empiryczne, których dokonują sieci neuronowe podczas swojej nauki, są nieznanymi zdaniami i nie mogą być ujęte jako hipotezy, to nie są falsyfikowalne. Jednocześnie, sieci neuronowe są dzięki temu łatwo weryfikowalne, bo bez trudu opanowują dane uczące i uzyskują wysokie wskaźniki trafności (nawet gdy są nierzetelne).

Rozwiązanie problemu, które autor tutaj proponuje, polega na ewaluacji sztucznych sieci neuronowych poprzez formułowanie hipotez ogólnych o ich rzetelności, które będą następnie poddawane falsyfikacji. Falsyfikacja opierać się będzie na obserwacjach, którymi będą poszczególne hipotezy statystyczne, dotyczące istotności różnic pomiędzy wynikami modelu testowanego w poszczególnych warunkach. Autor dokonuje więc tutaj syntezy falsyfikacyjnego i statystycznego testowania hipotez. Metodologia falsyfikacyjna polega na prawie rachunku zdań *modus tollendo tollens,* który zakłada, że jeżeli z hipotezy P wynika obserwacja q, a zaobserwowano, że q jest fałszywe, to hipoteza P musi być fałszywa. Natomiast, statystyczne testowanie hipotez sprawdza, czy różnica pomiędzy obserwacjami jest

234 K. Popper, *Logika odkrycia naukowego*, Warszawa 2002.

statystycznie istotna, *i.e.* czy prawdopodobieństwo losowego uzyskania takiej różnicy jest niewielkie. Jeżeli nie, to prawdziwa jest hipoteza zerowa H_0, a jeżeli tak, to prawdziwa jest hipoteza alternatywna H_1. Tak więc, podczas ewaluacji sieci neuronowych falsyfikowane będą hipotezy P o ich rzetelności, a obserwacjami q będą hipotezy statystyczne H odnoszące się do istotności różnic pomiędzy wskaźnikami osiąganymi przez te sieci w poszczególnych warunkach.

Modus tollendo tollens wyrażany jest przez poniższe równanie, z którego wynika, że jeżeli hipoteza P jest prawdziwa, to powinno się zaobserwować q, a że nie zaobserwowano q, to P musi być fałszywe.

$$((P \Rightarrow q) \wedge \neg q) \Rightarrow \neg P \qquad \text{(Równanie 5.0.1)}$$

Stwierdzić można w ten sposób, że jeżeli hipoteza P jest prawdziwa, to powinna zostać zaobserwowana hipoteza zerowa H_0, skoro jednak zaobserwowana została hipoteza alternatywna H_1 (bowiem $H_1 \equiv \neg H_0$), to hipoteza P musi być fałszywa.

$$((P \Rightarrow H_0) \wedge H_1) \Rightarrow \neg P \qquad \text{(Równanie 5.0.2)}$$

Podobnie ująć można, że jeżeli hipoteza zerowa H_0 jest prawdziwa, to zaobserwowana p-wartość (prawdopodobieństwo, że różnice pomiędzy obserwacjami są losowe) powinna być równa lub większa od krytycznego poziomu istotności α (na ogół $\alpha = 0.05$), skoro jednak zaobserwowana p-wartość jest niższa niż krytyczny poziom istotności α, to prawdą musi być hipoteza alternatywna H_1 (bowiem $H_1 \equiv \neg H_0$).

$$((H_0 \Rightarrow (p \geq \alpha)) \wedge (p < \alpha)) \Rightarrow H_1 \qquad \text{(Równanie 5.0.3)}$$

Falsyfikacja umożliwia oddzielenie nauki od nie-nauki, ponieważ *modus tollendo tollens* jest tautologią rachunki zdań (*i.e.* jest rozumowaniem niezawodnym). Weryfikacja polega zaś na rozumowaniu, które nie jest niezawodne, bowiem zakłada, że jeżeli hipoteza P jest prawdziwa, to powinno zostać zaobserwowane q, a zaobserwowano q, więc P jest prawdą.

$$((P \Rightarrow q) \wedge q) \Rightarrow P \qquad \text{(Równanie 5.0.4)}$$

Analogicznie stwierdzić można, że jeżeli P jest prawdziwe, to powinna zostać zaobserwowana hipoteza zerowa H_0, a ustalono że H_0 jest prawdą, więc P jest prawdą.

$$((P \Rightarrow H_0) \wedge H_0) \Rightarrow P \qquad \text{(Równanie 5.0.5)}$$

Powyższe rozumowanie weryfikacyjne nie uwzględnia jednak innych obserwacji (hipotez statystycznych), które również mogłyby wynikać z P, ale być fałszywe, dowodząc kategorycznie, że P jest fałszywe. Zawodność weryfikacji wynika stąd, że implikacja logiczna jest fałszywa wtedy i tylko wtedy, gdy hipoteza jest prawdziwa, a obserwacja jest fałszywa. Implikacja może być więc prawdziwa nawet wtedy, gdy hipoteza jest fałszywa, a obserwacja jest prawdziwa. Innymi słowy, prawdziwe obserwacje mogą uzasadniać fałszywe hipotezy.

Część empiryczna

Rozdział 6. Przykład ewaluacji sztucznych sieci neuronowych na przykładzie badań pismoznawczych.

6.1. Wprowadzenie. Opracowania na temat kryminalistycznych zastosowań sztucznych sieci neuronowych nie poruszają na ogół problemu ewaluacji tych metod ze względu na standardy dopuszczalności metod kryminalistycznych. Stosowne metody ewaluacji sztucznych sieci neuronowych powinny być zaś warunkiem koniecznym dla dopuszczalności ich zastosowania w kryminalistyce.

Ewaluacja polega na weryfikacji (pozytywna) i falsyfikacji (negatywna)[235]. Sieci neuronowe są nieinterpretowalne, bo ich procesy decyzyjne są nieznanymi zdaniami. Stąd pozytywne wnioskowania o ich rzetelności są nieużyteczne, zaś negatywne bardzo utrudnione. Innymi słowy, sieci neuronowe są bardzo łatwo weryfikowalne, skoro wystarczy podać ich ogólne wskaźniki trafności, zaś bardzo trudno falsyfikowalne, skoro niewiele można z nich wywnioskować. Podczas gdy, sieć neuronowa może być wysoce trafna, a jednocześnie zupełnie nierzetelna[236], *e.g.*: i) może kierować się wysoce dyskryminatywnymi cechami irracjonalnymi (takimi jak szum cyfrowy)[237]; ii) racjonalnymi cechami, które są wysoce dyskryminatywne na danym zbiorze danych, ale nie w rzeczywistości (takimi jak rodzaj narzędzia pisarskiego). Jedynym więc rozwiązaniem jest ewaluacja negatywna, gdzie: i) uda się odkryć nierzetelność modelu neuronowego, pomimo jego wysokich wyników testowych; ii) nie uda się odkryć nierzetelności modelu neuronowego, dowodząc możności jego zastosowań w praktyce.

Dlatego też celem kryminalistyki obliczeniowej powinno być opracowanie metod ewaluacji sieci neuronowych na potrzeby badań kryminalistycznych[238]. W tym celu autor przeprowadził przykładowe badania, gdzie: i) dokonywał prób falsyfikacji rzetelności; ii) dokonywał prób sanowania modeli neuronowych na podstawie udanych falsyfikacji. Przy tym, punktem wyjścia do badań były aksjomaty i metodyka badań

235 Ibid.

236 C. Rudin, *Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead*, „Nature Machine Intelligence" t. 1 nr 5 (2019), DOI: 10.1038/s42256-019-0048-x.

237 C. Zhang, S. Bengio, M. Hardt, B. Recht, O. Viwnyals, *Understanding Deep Learning (Still) Requires Rethinking Generalization*, „Communications of the ACM" t. 64 nr 3 (2021), DOI: 10.1145/3446776.

238 Tematyka ta została też częściowo przedstawiona w artykule: M. Marcinowski, *Evaluation of neural networks applied in forensics; handwriting verification example*, „Australian Journal of Forensic Sciences" (2022), DOI: 10.1080/00450618.2022.2079722.

pismoznawczych. Autor opracował kilkanaście modeli do weryfikacji wykonawstwa dokumentów odręcznych (obrazy statyczne), a przedstawił poniżej wyniki czterech najbardziej reprezentatywnych modeli.

Chcąc określić obecne metody ewaluacji sztucznych sieci neuronowych w dziedzinie kryminalistyki obliczeniowej, najlepiej kierować się największymi konkursami międzynarodowymi w tej dyscyplinie. Na przykład, podczas szesnastej edycji *International Conference on Document Analysis and Recognition* (ICDAR 2021), odbył się konkurs *On-Line Signature Verification Competition* (SVC 2021). Podczas tego konkursu porównywano modele obliczeniowe do weryfikacji odręcznych podpisów dynamicznych. W tym celu wyznaczono trzy zadania, wydzielone ze względu na jakość i metodę utrwalenia porównywanych podpisów, i.e.: i) podpisy rysikiem; ii) podpisy palcem; iii) podpisy rysikiem i palcem. Organizatorzy określili, że: „Każde zadanie będzie odrębnie ewaluowane, gdzie w każdym zadaniu przyznane będą trzy miejsca i odpowiadające im punkty [...] Miernikiem ewaluacji będzie popularny *Equal Error Rate* [lub *Crossover Error Rate*, *i.e*. punkt w którym wskaźniki błędów fałszywie pozytywnych i fałszywie negatywnych na krzywej ROC są równe [tłum. własne]”[239]. Jest to oczywiście użyteczna metoda, określająca trafność modelu w takim progu klasyfikacyjnym, dla którego wskaźniki klasowe są równe. Niemniej, jest to użyteczna metoda pozytywnej ewaluacji i porównywania trafności modeli, ale nie ich rzetelności, bo nie jest to metoda wystarczająca do odrzucenia nierzetelnego modelu. Ponadto, o ile metrykalne porównywanie modeli jest użyteczne z obliczeniowego punktu widzenia, to nie jest ono użyteczne z praktycznego punktu widzenia, gdzie ostatecznie lepszym będzie ten model, który rzetelniej sprawdzi się w rzeczywistości. Oczywiście, problem tkwi w tym, jak sprawdzić rzeczywistą skuteczność modelu.

Jak zaś wykazali autorzy niedawnego przeglądu metod automatycznej weryfikacji statycznych obrazów pisma (gdzie dominowały sztuczne sieci neuronowe), ewaluacja modeli sprowadza się do obliczenia zupełnie podstawowych wskaźników, takich jak trafność, czułość, czy swoistość[240].

239 International Association for Pattern Recognition, *International Conference on Document Analysis and Recognition 2021; Competition on On-line Signature Verification; SVC 2021* [na:] https://sites.google.com/view/SVC2021/home, dostęp 20 września 2021 r.

240 M.M. Hameed, R. Ahmad, M.L.M. Kiah, G. Murtaza, *Machine learning-based offline signature verification systems: A systematic review*, „Signal Processing: Image Communication” t. 93 (2021), DOI: 10.1016/j.image.2021.116139.

Rysunek 6.1.1. Schemat dziesięciokrotnej kroswalidacji, gdzie E oznacza średnią trafność lub błąd modelu, zaś E_i trafność lub błąd modelu w i-tym z dziesięciu kroków kroswalidacji.
Źródło: J. Ashfaque, A. Iqbal, *Introduction to Support Vector Machines and Kernel Methods* [na:] https://www.researchgate.net/publication/332370436_Introduction_to_Sus.ort_Vector_Machines_and_Kernel_Methods, dostęp 19 października 2021 r.

Popularną metodą porównywania modeli jest dziesięciokrotna kroswalidacja[241] (*10-fold cross-validation*; rys. 6.1.1), gdzie dla danego modelu: i) dzieli się zbiór danych na dziesięć równych części; ii) każda z tych części posłuży do przetestowania modelu, każdorazowo uczonego na pozostałych dziewięciu częściach; iii) uzyskuje się zatem dziesięć wariantów tego samego modelu, które różnią się doborem podzbiorów treningowego i testowego; iv) oblicza się średnią z trafności uzyskanych przez warianty modelu. Na ogół, aby porównać dwie różne architektury, obydwa modele poddaje się tej samej kroswalidacji, stąd: i) porównuje się średnie trafności obydwu kroswalidowanych modeli; ii) porównuje się trafności ich wariantów; iii) oblicza się istotność statystyczną zaobserwowanych różnic (na ogół będzie to test t-studenta dla zmiennych zależnych [*paired t-test*], gdzie warianty obydwu modeli porównywane są parami)[242]. Niestety, metoda ta nadal sprowadza się do metrykalnego porównania modeli, które nie przesądza o ich rzetelności w warunkach rzeczywistych.

241 W dziedzinie uczenia maszynowego, walidacja (*validation*) oznacza test modelu przeprowadzony na podstawie danych, które nie służyły do nauki tego modelu, chociaż pochodziły z tego samego zbioru uczącego, z którego pochodziły dane uczące model. Na przykład, model uczony rozróżniania ludzi na podstawie fotografii ich twarzy, może być walidowany na fotografiach osoby, której wizerunek służył już do uczenia modelu, pod warunkiem, że nie będą to te same fotografie.

242 T.G. Dietterich, *Approximate Statistical Tests for Comparing Supervised Classification Learning Algorithms*, „Neural Computation” t. 10 (1998).

W opinii autora, należy oczywiście wspierać badania nad sieciami neuronowymi do zastosowań kryminalistycznych. Jednakże, trudno aby stanowiły one rozwiązania problemów kryminalistycznych, skoro brak jest metod ewaluacji umożliwiających dowodzenie takich twierdzeń. Podobnie, trudno aby znalazły one zastosowanie w praktyce, skoro brak jest procedur dla ich zastosowań, szczególnie że procedury muszą znajdować uzasadnienie w twierdzeniach empirycznych. Badania nad metodyką ewaluacji są więc kluczowe, stąd ważne jest aby do nich zachęcać, oraz poszerzać w tym zakresie wiedzę ekspertów, którzy nie zawsze są świadomi pułapek uczenia maszynowego. Na przykład, badacze z renomowanego *National Institute of Standards and Technology* (NIST) wypowiedzieli się bardzo zasadnie na temat automatycznych metod badania dokumentów i ich ewaluacji, zauważając jednocześnie, że: „Rezultaty zastosowania [modelu CEDAR-FOX[243]] pokazały ponad 95% trafność, która wystarczyłaby dla dopuszczenia opinii pismoznawczej w sprawach Dauberta i Frye'a”[244]. Należało tutaj zastrzec, że chociaż standardy Dauberta i Frye'a wydawać by się mogły spełnione, to tylko pozornie, bo nie określono rzetelności modelu.

Należy wspomnieć, że istnieje niezwykle bogata literatura statystyczna, dotycząca *i.a*. automatycznej ewaluacji i klasyfikacji dowodów w kryminalistyce[245], oraz ewaluacji metod obliczeniowych[246], ale przede wszystkim probabilistycznych i

243 M. Goc, *Współczesny model ekspertyzy pismoznawczej; wykorzystanie nowych metod i technik badawczych*, Warszwa - Szczecin 2015, s. 242–243.

244 M. Taylor, C. Bird, B. Bishop, T. Burkes, M.P. Caligiuri, B. Found, W.P. Grose, L.R. Logan, K.E. Melson, M.L. Merlino, L.S. Miller, L. Mohammed, J. Morris, J.P. Osborn, N. Osborne, B. Ostrum, C.P. Saunders, S.A. Shappell, H.D. Sheets, S.N. Srihari, R.D. Stoel, T.W. Vastrick, H.E. Waltke, E.J. Will, *Forensic Handwriting Examination and Human Factors: Improving the Practice Through a Systems Approach*, Gaithersburg 2020, s. 70.

245 S. Bozza, F. Taroni, R. Marquis, M. Schmittbuhl, *Probabilistic evaluation of handwriting evidence: likelihood ratio for authorship*, „Applied Statistics” t. 57 (2008); A. Crawford, A. DM, S. CP, *Bayesian hierarchical modelling for the forensic evaluation of handwritten documents*, „Law, Probability and Risk” t. 17 (2020); N. Garton, D.M. Ommen, J. Niemi, A. Carriquiry, *Score-based likelihood ratios to evaluate forensic pattern evidence* [w:] 2020; D.M. Ommen, C.P. Saunders, *Building a unified statistical framework for the forensic identification of source problems*, „Law, Probability and Risk” t. 17 nr 2 (2018), DOI: 10.1093/lpr/mgy008; F. Taroni, S. Bozza, A. Biedermann, P. Garbolino, C. Aitken, *Data Analysis in Forensic Science; A Bayesian Decision Perspective*, Chichester 2010.

246 A. Brink, L. Schomaker, M. Bulacu, *Towards Explainable Writer Verification and Identification Using Vantage Writers* [w:] *W: Ninth International Conference on Document Analysis and Recognition (ICDAR 2007*, Washington 2007; C. Fuglsby, C.P. Saunders, *U-statistics for estimating performance metrics in forensic handwriting analysis*, „Journal of Statistical Computation and Simulation” t. 90 nr 6 (2020); I. Guyon, L. Schomaker, R. Plamondon, M. Liberman, S. Janet, *UNIPEN project of on-line data exchange and recognizer benchmarks* [w:] *W: Proceedings of the 12th IAPR International Conference on Pattern Recognition*, Washington 1994.

statystycznych[247]. Są to metody często komplementarne wobec problemu ewaluacji sztucznych sieci neuronowych, ale nie dotyczą go bezpośrednio.

Ewaluacja nie jest szczególnie popularnym przedmiotem badań i literatury na temat uczenia maszynowego, a sprowadzana jest na ogół do zagadnienia interpretowalności[248]. Warto tutaj wyróżnić artykuł Bidermana i Scheirera z 2021 roku[249], którzy wyróżniają najczęstsze uchybienia ewaluacyjne: i) brak rygorystycznej ewaluacji statystycznej, a przede wszystkim określeń istotności statystycznej wyników; ii) brak zerowych modeli porównawczych (*null models*), *i.e.* struktur analogicznych do hipotezy zerowej (*null hypothesis*), *e.g.* modeli o losowych parametrach, które pozwoliłyby określić z jakim prawdopodobieństwem dana trafność mogła być osiągnięta losowo; iii) brak zewnętrznej ewaluacji (w tym przez recenzentów), oraz podwójna nieinterpretowalność powodowana przez nietransparentność metod i modeli komercyjnych. Autorzy zaproponowali też dobre praktyki, które pomóc by mogły w dziedzinie: i) statystyczne sprawdzanie i opisywanie wyników (e.g. podawanie średniej i odchylenia standardowego wyników uzyskanych podczas kroswalidacji); ii) tworzenie modeli kontrolnych, uczonych klasyfikacji danych ze względu na ich źródłowy zbiór danych; iii) ewaluacja dokonywana przez osoby trzecie.

6.2. Metody.

Dane. Zbiór 1604 odręcznie pisanych dokumentów wykonanych przez 310 osób (skany jednostronnych dokumentów A4, obrazy statyczne) pobrany został z bazy CVL (*Computer Vision Lab, Institute of Computer Aided Automation, Vienna*

247 G. Filipe, P. Correia, D. Meuwly, D. Vloed, *Empirical validation of likelihood ratio methods; a case study in forensic speaker recognition* [w:] *W: 2016 4th International Conference on Biometrics and Forensics (IWBF). IEEE, Washington*, 2016; R. Haraksim, D. Ramos, D. Meuwly, *Validation of likelihood ratio methods for forensic evidence evaluation handling multimodal score distributions*, „IET Biometrics" t. 6 (2017); R. Haraksim, D. Ramos, D. Meuwly, C.E.H. Berger, *Measuring coherence of computer-assisted likelihood ratio methods*, „Forensic Science International" t. 249 (2015); D. Ramos, J. Gonzalez-Rodriguez, *Reliable support: Measuring calibration of likelihood ratios*, „Forensic Science International" t. 230 (2013); D. Ramos, R. Haraksim, D. Meuwly, *Likelihood ratio data to report the validation of a forensic fingerprint evaluation method*, „Data in Brief" t. 10 (2017); A. Tauseef, L. Spreeuwers, R. Veldhuis, D. Meuwly, *Biometric evidence evaluation: an empirical assessment of the effect of different training data*, „IET Biometrics" t. 3 (2014).

248 C. Rudin, *Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead*, „Nature Machine Intelligence" t. 1 nr 5 (2019), DOI: 10.1038/s42256-019-0048-x.

249 S. Biderman, W.J. Scheirer, *Pitfalls in Machine Learning Research: Reexamining the Development Cycle*, arXiv, 18 sierpnia 2021 r., http://arxiv.org/abs/2011.02832.

University of Technology)[250]. Zbiór 1539 odręcznie pisanych dokumentów wykonanych przez 657 osób (skany jednostronnych dokumentów A4, obrazy statyczne) pobrany został z bazy IAM (*Institut für Informatik und Angewandte Mathematik, University of Bern*)[251]. Stąd połączony zbiór treningowy wynosił 2740 obrazów dokumentów (1415 z CVL i 1325 z IAM) wykonawstwa 822 osób (283 z CVL i 539 z IAM). Połączony zbiór testowy wynosił 403 obrazy dokumentów (189 z CVL i 214 z IAM) wykonawstwa 145 osób (27 z CVL i 118 z IAM).

W przypadku zbioru IAM: i) wzory tekstów pochodziły z korpusu LOB (*Lancaster - Oslo/Bergen corpus*) zawierającego 500 tekstów w języku angielskim, każdy liczący około 2000 słów (tab. 6.2.1); ii) teksty z korpusu dzielono na fragmenty liczące od trzech do sześciu zdań, oraz minimum 50 słów; iii) formularze drukowano na czystych kartkach A4, pod którymi umieszczono kartki liniowane (1.5 cm interlinii); iv) probantów proszono o skopiowanie wzorców swoim „codziennym pismem" i narzędziem pisarskim, oraz o zaprzestanie pisania w przypadku wyczerpania wyznaczonej przestrzeni pisarskiej; v) dokumenty skanowano za pomocą HP-Scanjet 6100 w standardzie 300 dpi, zapisując w 8 bitowej skala szarości, format TIFF.

Identyfikator	Kategoria tekstu	Liczba tekstów
A	Reportaż prasowy	44
B	Edytorial prasowy	27
C	Recenzja Prasowa	17
D	Religijny	17
E	Hobbystyczny	38
F	Folklor	44
G	Biografie i eseje	77
H	Różnorodne	30
J	Naukowy	80
K	Fikcja	29
L	Kryminał	24
M	Fikcja Naukowa	6
N	Przygodowy	29
P	Romans	29
R	Humor	9
	Suma	500

Tabela 6.2.1. Kategorie wzorców tekstu z korpusu LOB.

Źródło: U. Marti, H. Bunke, *The IAM-database: An English Sentence Database for Off-line Handwriting Recognition*, „International Journal on Document Analysis and Recognition" t. 5 (2002).

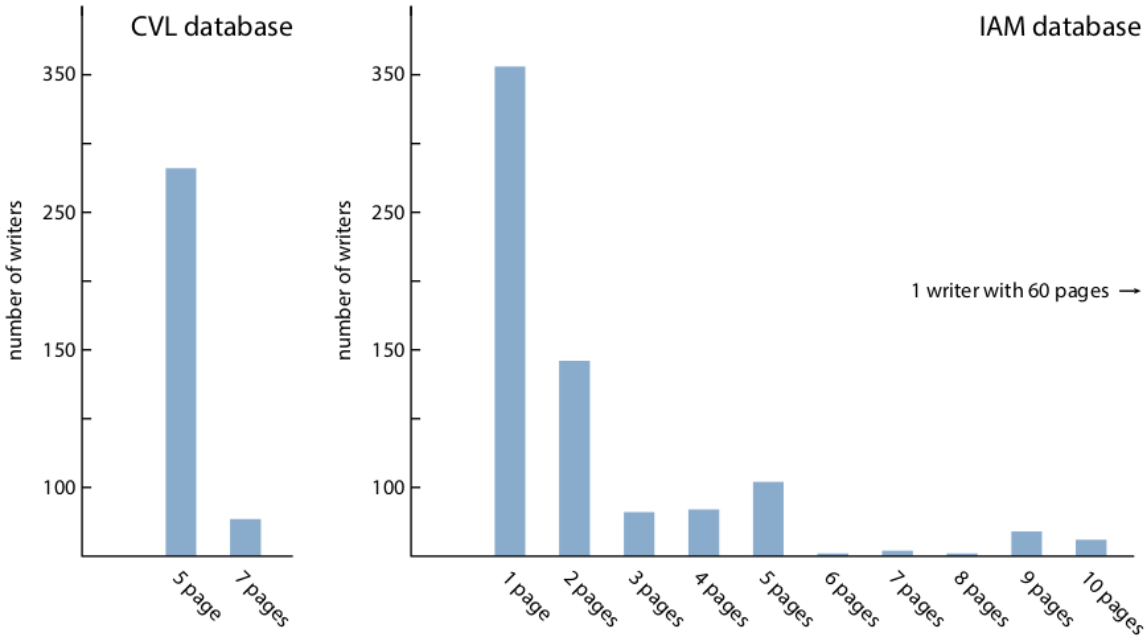
250 F. Kleber, S. Fiel, M. Diem, R. Sablatnig, *CVL-DataBase: An Off-Line Database for Writer Retrieval, Writer Identification and Word Spotting* [w:] *2013 12th International Conference on Document Analysis and Recognition*, Washington, DC, USA 2013.

251 U. Marti, H. Bunke, *The IAM-database: An English Sentence Database for Off-line Handwriting Recognition*, „International Journal on Document Analysis and Recognition" t. 5 (2002).

W przypadku zbioru CVL: i) wyznaczono 7 wzorców tekstu w języku angielskim i niemieckim (tab. 6.2.2); ii) 27 probantów wykonało po jednym dokumencie na każdy z 7 wzorców (wzorce nr 1–8; podzbiór testowy); iii) 284 probantów wykonało po jednym dokumencie na każdy z 5 wzorców (wzorce nr 1–6; podzbiór treningowy); iv) wobec probantów zastosowano takie same formularze i standardy jak w przypadku zbioru IAM; v) dokumenty skanowano za pomocą Lexmark X652de, standard 300 dpi, zapisywano w 24 bitowej skali kolorów RGB, format TIFF.

Identyfikator	Autor	Tytuł	Liczba słów
1	Edwin A. Abbot-Flatland	A Romance of Many Dimension	90
2	William Shakespeare	Mac Beth	47
3	**Wikipedia**	**Mailüfterl**	**74**
4	Charles Darwin	Origin of Species	52
6	**Johann Wolfgang von Goethe**	**Faust. Eine Tragödie**	**50**
7	Oscar Wilde	The Picture of Dorian Gray	65
8	Edgar Allan Poe	The Fall of the House of Usher	73

Tabela 6.2.2. Wzorce tekstów z bazy CVL (wytłuszczeniem oznaczono teksty w języku niemieckim). Źródło: F. Kleber, S. Fiel, M. Diem, R. Sablatnig, *CVL-DataBase: An Off-Line Database for Writer Retrieval, Writer Identification and Word Spotting* [w:] *2013 12th International Conference on Document Analysis and Recognition*, Washington, DC, USA 2013, s. 560–561.



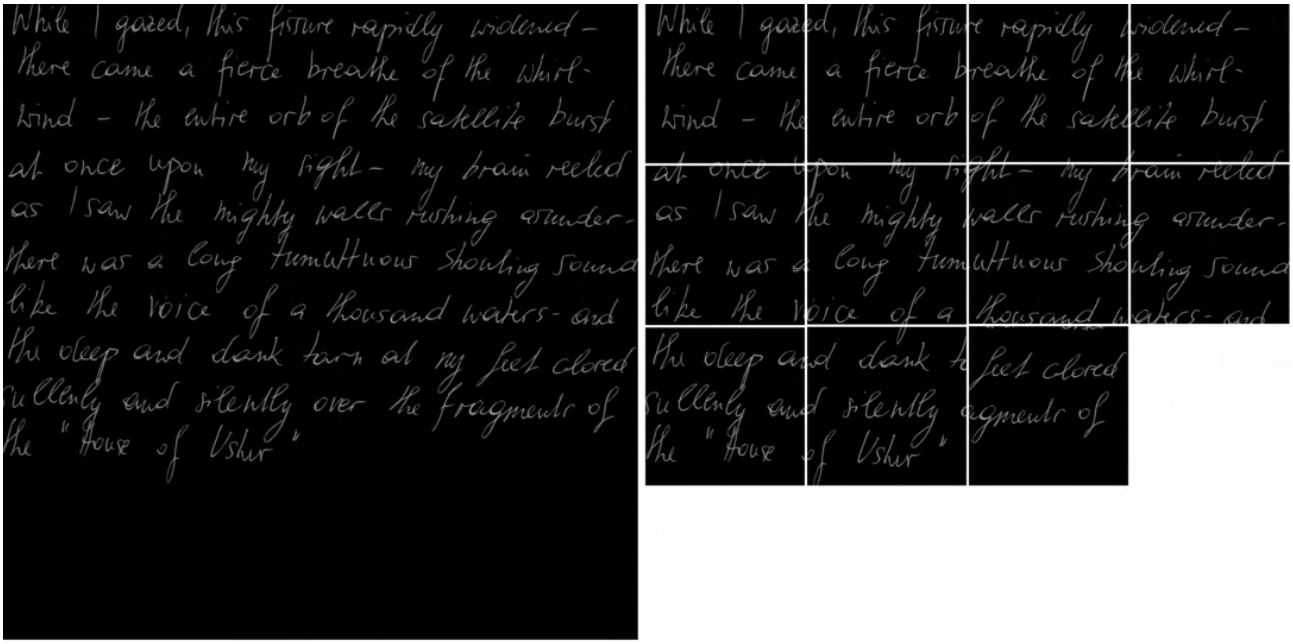
Rysunek 6.2.1. Dystrybucja probantów ze względu na liczbę wykonanych próbek dla zbioru CVL i IAM. Źródło: F. Kleber, S. Fiel, M. Diem, R. Sablatnig, *CVL-DataBase: An Off-Line Database for Writer Retrieval, Writer Identification and Word Spotting* [w:] *2013 12th International Conference on Document Analysis and Recognition*, Washington, DC, USA 2013.

Autorzy zbioru CVL opracowali też wykres ilustrujący dystrybucję probantów ze względu na liczbę sporządzonych dokumentów, zarówno dla zbioru CVL i IAM (rys. 6.2.1). Wyjaśnić w związku z tym należy, że autor przeprowadził uczenie modeli neuronowych na fragmentach obrazów z tych baz, a nie na całych obrazach. Pierwszym powodem było obniżenie kosztów obliczeniowych związane z obniżeniem rozmiaru przetwarzanych danych. Drugim powodem było to, że większość probantów z bazy IAM wykonało zaledwie po jednym dokumencie. Gdyby bowiem uczyć model na całych obrazach dokumentów, to dokumentów tych nie można by było wykorzystać przy problemie weryfikacyjnym (gdzie porównywane są dwie próbki), ponieważ w tych przypadkach, zadaniem modelu byłoby stwierdzenie, że dwa identyczne obrazy pochodzą od tego samego wykonawcy.

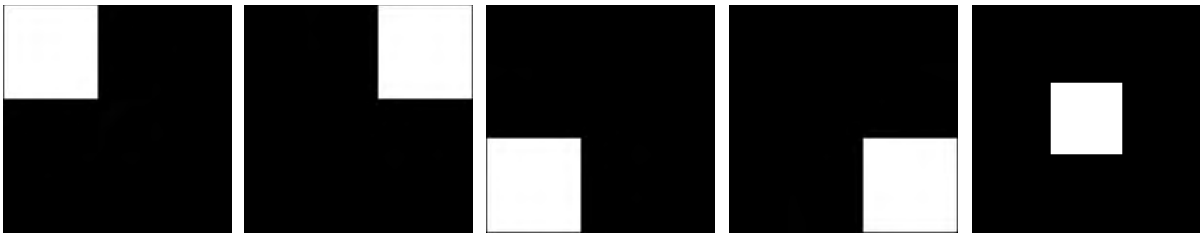
Preprocesowanie. Obrazy dokumentów były przekształcane do skali szarości lub binaryzowane, następnie przeprowadzano inwersję kolorów, ekstrakcję przestrzeni pisarskiej o wymiarach 2048 x 2048 px, redukcję rozmiaru ekstraktów do wymiarów 1024 x 1024 px, podział ekstraktów na fragmenty o wymiarach 256 x 256 px (rys. 6.2.2), konwersję z formatu TIFF do PNG.

Fragmenty, które nie zawierały tekstu lub zawierały małe jego ilości były odrzucane. W przypadku modeli v1.1.0, v2.1.0 i v2.4.0, filtrowanie fragmentów polegało na progowaniu ze względu na średnią wartość pikseli. W przypadku modelu v2.5.1. zastosowano bardziej subtelną i skuteczną metodę. Otóż, skoro dokonywano uprzedniej inwersji kolorów, to kolor tła był czarny zamiast białego, takie zaś piksele posiadają wartość zero, a suma pikseli na obrazie pozbawionym tekstu również wynosi zero. Stąd, na potrzeby selekcji fragmentów: i) odszumiano fragmenty poprzez progowanie wartości pikseli niższych niż 55 do zera; ii) wykonywano iloczyn fragmentu z czarno-białym filtrem o tym samym rozmiarze i sumowano uzyskane wartości; iii) jeżeli wynik sumy dla wszystkich filtrów był większy od zera to fragmentu nie pomijano. Filtry opracowano aby zapewnić wykorzystanie takich tylko fragmentów obrazów, gdzie tekst zachodził na wszystkie białe pola (rys. 6.2.3).

Odnotować musimy, że odszumianie fragmentów podczas ich filtrowania wykorzystywano tylko w tym celu. Jedynie model v2.4.0 uczony był na obrazach odszumianych poprzez zerowanie pikseli o wartościach niższych niż 25.



Rysunek 6.2.2. Przykładowy ekstrakt (po lewej) i jego fragmenty (po prawej) po preprocesowaniu.
Źródło: opracowanie własne.



Rysunek 6.2.3. Filtry zastosowane do selekcji fragmentów obrazów.
Źródło: opracowanie własne.

Arkusze danych. Ścieżki dostępu, nazwy i przynależności klasowe par obrazów zawarto w arkuszach danych (*dataframes*). Utworzone zostały z założeniem, że liczba pozytywnych i negatywnych przypadków powinna być równa (*i.e.* klasy powinny być równoliczne). Ponieważ maksymalna liczba negatywnych kombinacji była znacznie większa niż maksymalna liczba pozytywnych kombinacji, to: i) utworzono najpierw wszystkie możliwe pary pozytywne; ii) wylosowano taką liczbę par negatywnych, która równa była liczbie par pozytywnych. Należy odnotować, że dla danej pozytywnej pary xy, utworzona została także pozytywna para yx. Podczas gdy, w przypadku klasy negatywnej było to mało prawdopodobne, aby wylosować daną parę i jej odwrotność. Tak więc, w sensie ilościowym obydwie klasy były równe, ale w sensie jakościowym klasa negatywna była prawie dwukrotnie większa od pozytywnej.

W przypadku modeli uczonych na połączonych bazach CVL i IAM: i) dla modelu v2.4.0 dopuszczono generowanie par obrazów pomiędzy bazami danych (*i.e.* gdzie obraz x pochodzi z bazy CVL zaś obraz y z bazy IAM), przyjęto przy tym, że wszystkie takie pary są negatywne); ii) dla modelu v2.5.1 wykluczono takie przypadki.

Modele. Wszystkie poniższe modele (v1.1.0, v2.1.0, v2.4.0, v2.5.1) stanowiły dwuścieżkowe sieci konwolucyjne, gdzie obydwie ścieżki stanowią identyczne ale odrębne sieci konwolucyjne, a każda z nich przetwarza jeden z obrazów stanowiących parę, ekstraktując jego cechy i prezentując je jako wektor cech. Pary wektorów cech trafiają następnie do warstw w pełni połączonych, gdzie dokonywana jest ostateczna klasyfikacja par obrazów (do klasy „ten sam" albo „różni wykonawcy").

Pośród cech wspólnych architektury, wszystkie poniższe modele: i) posiadały trzy warstwy konwolucyjne, liczące kolejno 256, 512 i 1024 filtry, których rozmiar wynosił kolejno 12 x 12, 6 x 6 i 3 x 3 px, a krok 4, 2, 1 (tak, aby w efektywnym polu receptywnym znajdowały się całe litery); ii) po warstwach konwolucyjnych następowały normalizujące, a dalej redukujące poprzez wyciągnięcie najwyżej wartości (wymiar okna 2 x 2 px, krok 2); iii) ostatnia warstwa redukująca wyciągała średnią globalną z map aktywności, skutkując wektorem 1024 cech; iv) wektory przekazywane były do warstw w pełni połączonych, liczących kolejno 1024, 512, 256 i 1 neuron; v) przed każdą warstwą w pełni połączoną znajdowała się warstwa opuszczająca (prawdopodobieństwo wyzerowania 0.5), a po każdej z nich znajdowała się warstwa normalizująca (*batch-normalization*); vi) obliczano dystanse euklidesowy i kosinusowy dla każdej pary wektorów cech (ekstraktowanych przez ścieżki konwolucyjne); vii) sygnał wyjściowy z warstw w pełni połączonych (*i.e.* z warstwy czwartej) oraz dystanse przekazywano do neuronu wyjściowego, który dokonywał ostatecznej klasyfikacji; viii) jako funkcję aktywacji zastosowano wszędzie ReLU, za wyjątkiem ostatniego neuronu, gdzie była to funkcja sigmoidalna.

Jedynie model v1.1.0 różnił się istotnie pod względem architektury od pozostałych, otóż: i) nie obliczano dystansów, więc klasyfikacji dokonywał neuron czwartej warstwy gęstej (aktywowany przez funkcję sigmoidalną); ii) warstwa redukująca poprzez wyciągnięcie średniej poprzedzona była przez warstwę głosującą; iii) rozmiar filtrów wynosił 11 x 11, 5 x 5 i 3 x 3 px, zaś krok kolejno 4, 1, 1.

Pod względem parametrów, zastosowano: i) optymizator Adam (*Adaptive Moment Estimation*) z rekomendowanymi parametrami[252]; ii) redukcje współczynnika uczenia następowały arbitralnie, współczynnik redukcji wynosił 0.1 lub 0.01 (wykonywano iloczyn współczynnika uczenia i redukcji); iii) *batch-size* wynosił 16; iv) funkcję kosztu stanowiła binarna entropia krzyżowa (*binary cross-entropy*)[253].

Podstawową różnicą między modelami był sposób ich uczenia, otóż: i) modele v1.1.0 i v2.1.0 trenowane były na surowych obrazach w skali szarości, pochodzących z bazy CVL; ii) model v2.4.0 trenowany był na obrazach binaryzowanych i odszumianych (przez progowanie wartości pikseli niższych niż 25 do 0), które pochodziły z bazy CVL i IAM; iii) model v2.5.1 trenowany był na obrazach w skali szarości, pochodzących z bazy CVL i IAM, których nie poddawano żadnym dodatkowym zabiegom.

Ponieważ model v2.5.1 uzyskał najwyższą trafności, stanowi więc przykład dominujący.

Metody ewaluacji. Obrano dwa podejścia do ewaluacji, *i.e.*: i) ewaluację opartą na wyznaczaniu podzbiorów, gdzie dokonuje się podziału zbioru testowego na mniejsze podzbiory ze względu na właściwości tych podzbiorów, źródeł danych i baz danych; ii) ewaluację opartą na manipulacjach danymi wejściowymi, gdzie modyfikuje się dane wejściowe ze względu na aksjomaty i metodykę badań pismoznawczych. Istota powyższego rozróżnienia jest taka, że wyniki ewaluacji opartej na modyfikacjach danych wejściowych, rozbijać można ze względu na podzbiory.

Metody oparte na podzbiorach.

Kryteria. Model v2.4.0 (trenowany na binaryzowanych i odszumianych obrazach) oraz v2.5.1 (trenowany na surowych obrazach w skali szarości) uczone były na połączonych zbiorach CVL i IAM, jest to więc „Podstawowe” kryterium ewaluacji, po którym następują podzbiory/kryteria „IAM” i „CVL”. Zbiory testowe wyznaczano tak, aby nie zawierały tych samych dokumentów, wykonawców i wzorców tekstu co zbiory treningowe. Jednakże, zbiór testowy CVL pokrywał się z treningowym pod

252 D.P. Kingma, J. Ba, *Adam: A Method for Stochastic Optimization*, arXiv, 29 stycznia 2017 r., http://arxiv.org/abs/1412.6980.
253 A. Glassner, *Deep Learning: From Basics to Practice*, Seattle 2018, s. 1164–1165.

względem wzorców tekstu (pokrywały się nr 1–6; nie pokrywały nr 7 i 8). Wyznaczono zatem taki podzbiór CVL, który był rozbieżny ze zbiorem treningowym CVL, zarówno pod względem wykonawców jak i wzorców tekstu, *i.e.* „CVL-Rozłączny".

Kryterium „Negatywne-Surowe" wyznaczono aby określić podzbiór przypadków negatywnych zachodzących pomiędzy bazami CVL i IAM. Kryterium „Negatywne-Odszumione" wyznaczono dla takich par obrazów, które zachodzą na pomiędzy bazami CVL i IAM, oraz zostały odszumione poprzez zerowanie wartości pikseli niższych niż 55. Dla modelu v2.4.0 zastosowano tylko „Negatywne" kryterium, ponieważ uczony był już i testowany na binaryzowanych oraz odszumianych obrazach (próg wartości pikseli określony na 25).

Podczas gdy żadne pary obrazów zachodzących pomiędzy bazami danych nie zostały wykorzystane do uczenia modelu v2.5.1, a dane uczące modelu v2.4.0 zawierały takie przypadki. To dla zapewnienia porównywalności wyników, wyznaczono kryterium „Średniej" z rezultatów osiągniętych przez model v2.4.0 wobec odrębnych kryteriów IAM i CVL. Podobnie, wyznaczono kryterium „Średniej" z rezultatów osiągniętych przez model v2.5.1 wobec kryterium Podstawowego i Negatywnego-Surowego, aby sprawdzić w jakim stopniu jego rezultaty mogły by zostać zawyżone.

Kategorie. Jednym z podstawowych sposobów ewaluacji metod biometrycznych jest testowanie ich ze względu na właściwości źródeł danych, szczególnie ze względu na kategorie statystyczne. Jednakże, bazy IAM ani CVL nie zawierały stosownych informacji, stąd autor postanowił dokonać własnej kategoryzacji probantów, zaliczając ich do zbiorów cech pisma (a nie do kategorii *per se*), które uważane bywają za wskaźniki – kobiecości, męskości, leworęczności i praworęczności pisma – wymieniane przede wszystkim przez R. A. Hubera *et. al*[254]. Na zbiór cech pisma kobiecego składały się: i) wysokie wyrobienie, staranność, czytelność i dojrzałość pisma; ii) ogólny obraz pisma owalny lub okrągły; iii) jednolite nachylenie pisma; iv) manieryzmy. Na zbiór cech pisma męskiego składały się: i) niskie wyrobienie, staranność, czytelność i dojrzałość pisma; ii) pismo kątowe; iii) nachylenie głębsze niż 70 stopni; iv) kreskowe znaki diakrytyczne nad „i"; v) duża siła nacisku lub cieniowania; vi) pismo drobne, duże lub bardzo duże. Na zbiór cech pisma

254 R.A. Huber, A.M. Headrick, H.H. Harralson, L.S. Miler, *Handwriting Identification Facts and Fundamentals*, Boca Raton 2018.

leworęcznego składały się: i) elementy poziome kreślone w kierunku wstecznym; ii) owale, łuki i pętlice kreślone zgodnie z ruchem wskazówek zegara; iii) pismo pionowe lub lewoskośne; iv) pismo o zmiennym nachyleniu; v) terminacje do góry i w lewo; vi) nacisk wstępujący. Wskaźnikiem przynależności do zbioru cech pisma praworęcznego był brak silnych wskazań przynależności do zbioru cech pisma leworęcznego.

Metody oparte na danych wejściowych.

Różne ilości danych. Jednym z najważniejszych standardów w dziedzinie badań pismoznawczych jest wymóg zapewnienia odpowiedniej ilości materiału porównawczego i kwestionowanego. Gdzie, większa ilość danych gwarantuje zawsze bardziej pewne rezultaty, zaś minimum ilościowe zależy często od jakości danych.

Niekiedy, według tego standardu definiowana jest solidność (*robustness*) metod probabilistycznych (często tożsama z rzetelnością, *reliability*). Haraksim *et al.* przyjmują, że: „Solidność definiowana jest jako zdolność metody do utrzymania wyników pomimo zmniejszania ilości danych [tłum. własne]”[255].

Z punktu widzenia przyjętej przez autora definicji rzetelności, powyższa definicja solidności jest jej szczególnym przypadkiem. Autor dążył jednak do sprawdzenia czy trafność modelu v2.5.1 zostanie utrzymana lub wzrośnie w wyniku zwiększenia ilości danych wejściowych. Stąd, testowano model poprzez wprowadzanie do niego ekstraktów całej przestrzeni pisarskiej (przeskalowanych z 2048 x 2048 px do 1024 x 1024 px).

Różne jakości danych. Ponieważ zdarzyć się może, że dokument kwestionowany lub materiał porównawczy sporządzony jest na papierze kratkowanym lub liniowanym, model powinien być zdolny uwzględnić wpływ tych okoliczności na obraz nawyku pisarskiego. W rzeczywistości jednak, model przetestować należy ze względu na sam fakt występowania kratek lub linii, skoro ani baza IAM ani CVL nie zawierała dokumentów sporządzonych na takowym papierze.

Nanoszono więc na obrazy pisma kratkę analogiczną do typowej kratki papierowej (rys. 6.2.4). Gdzie: i) wartość pikseli wynosiła 54 (tuż pod progiem odszumiania obrazów dla modelu v2.5.1); ii) przy standardzie skanów 300 dpi (*i.e*. 300

255 R. Haraksim, D. Meuwly, P. Vergeer, *Fingerprint Evidence Evaluation, Robustness to the Lack of Data* [w:] Netherlands Forensic Institute, Hague 2012.

pikseli na cal wydruku), wycinek przestrzeni pisarskiej o wymiarach 2048 x 2048 px jest równoważny 17.34 x 17.34 cm ekstraktom i 4.34 x 4.34 cm fragmentom, stąd, przy założeniu 0.5 cm kratki, fragment powinien liczyć 8.5 kratek w kierunku wertykalnym i horyzontalnym; iii) wszystkie kratki posiadały jednakową grubość 2 px; iv) kratkę wygenerowano w rozmiarze 256 x 256 px i nanoszono poprzez dodawanie do preprocesowanych już fragmentów obrazów.

Rysunek 6.2.4. Przykładowa kratka nanoszona na fragmenty obrazów z bazy CVL i IAM.
Źródło: opracowanie własne.

Różna trudność danych. Oceniając rzetelność metod kryminalistycznych, ewaluacji dokonywać należy na różnych stopniach trudności problemu. Aby dokonać ewaluacji modelu v2.5.1 z pominięciem najprostszych, a zarazem irrelewantnych problemów, to pominąć należy pary identycznych fragmentów. W dalszej kolejności rozważać można pominięcie takich par fragmentów, które pochodzą z tego samego dokumentu, a nawet takich par fragmentów, które sporządzono w jednakowym języku.

Na pierwszym i drugim poziomie trudności, rezultaty dla kryterium Negatywnego byłyby równoważne ogólnym rezultatom kryterium Negatywnego (skoro w obydwu przypadkach porównywane fragmenty nie są identyczne i nie pochodzą z tego samego dokumenty), więc zostały pominięte. Ponadto, na drugim poziomie trudności, istotnie ograniczony był udział par z bazy IAM, gdzie większość probantów napisała po jednym dokumencie. Na trzecim poziomie trudności, ponieważ baza IAM zawierała teksty w tylko języku angielskim, to pary różnojęzyczne byłyby zawsze

negatywne, zachodząc pomiędzy bazami danych (pary anglojęzycznych dokumentów z IAM i niemieckojęzycznych z CVL), stąd sztucznie zniekształcałyby wyniki. Natomiast, wyróżniono je jako kryterium Negatywne, gdzie wszystkie przypadki z bazy IAM parowane były z wzorcami nr 3 i 6 bazy CVL. Tak więc, w kryterium Podstawowym dla trzeciego poziomu uwzględniono tylko dokumenty z bazy CVL (gdzie parowano wzorce nr 1, 2, 4, 7, 8 z wzorcami nr 3, 6). Natomiast osłabione tutaj kryterium CVL-Rozłączne polegało na parowaniu wzorców nr 7 i 8 z wzorcami nr 3 i 6.

Powyższe podejście stanowi metodę ewaluacji poprzez wyznaczanie podzbioru, ale autor postanowił zaliczyć ją tutaj, skoro wyniki niniejszej metody przedstawione zostaną ogólnie oraz rozbite ze względu na kryteria.

Różne metody preprocesowania. Podejrzewać można, że niektóre modele rozróżniają wykonawców na podstawie instrumentów pisarskich (zakładając, że różni wykonawcy używali różnych narzędzi pisarskich i używali ich systematycznie). Stąd, autor zdecydował przetestować model v2.5.1 (trenowany na surowych obrazach w skali szarości) na podstawie binaryzowanych i odszumianych obrazów (*vide* model v2.4.0, gdzie próg odszumiania wynosił 25).

Różne źródła danych. Jakakolwiek metoda, aby przyjąć że jest rzetelna, powinna dawać rezultaty analogiczne kiedy testowana jest na analogicznych danych. Stąd, modele v1.1.0 i v2.1.0, których architektury różniły się jedynie nieznacznie, a które trenowane były na zbiorach obrazów z bazy CVL (surowe obrazy w skali szarości), poddane zostały ewaluacji na bazie IAM (również surowych obrazów w skali szarości), zawierającej skany gorszej jakości (wysoce zanieczyszczone szumem).

Istotność statystyczna. Celem określenia statystycznej istotności różnic zachodzących między wynikami, przeprowadzenie testów statystycznych nie byłoby wystarczające, ponieważ zmienne te nie są niezależne[256].

Ponieważ różnice w wynikach zachodzą pomiędzy danym zbiorem a jego podzbiorami, to ustalić należy z jakim prawdopodobieństwem wyniki uzyskane dla danego podzbioru mogą też wystąpić dla losowych podzbiorów. Jeżeli wynik uzyskany

256 T.G. Dietterich, *Approximate Statistical Tests for Comparing Supervised Classification Learning Algorithms*, „Neural Computation" t. 10 (1998).

dla danego podzbioru występuje losowo z wysokim prawdopodobieństwem, to uzyskane różnice nie są istotne. Najprostszą metodą sprawdzenia istotności uzyskanych wyników, będzie: i) na danym zbiorze losowo wyznaczyć daną liczbę podzbiorów o określonych właściwościach (*e.g*. liczbie par dokumentów lub wykonawców od których dokumenty pochodzą); ii) przetestować model na tych podzbiorach; iii) określić dyskretną dystrybucję prawdopodobieństwa uzyskania takich rezultatów.

Na przykład, jeżeli zbiór testowy liczy 1 mln par dokumentów, a dany jego podzbiór liczy 1000 par dokumentów, i wystąpiła rozbieżność w wynikach modelu dla tych zbiorów, to zasadnym jest pytanie, czy różnica ta jest dowodem nierzetelności modelu, czy może jest ona przypadkowa (nieistotna statystycznie)? Można więc na zbiorze 1 mln par dokumentów wyznaczyć losowo 10 000 podzbiorów liczących 1000 par dokumentów każdy. Następnie przetestować model na tych podzbiorach i uzyskać 10 000 wyników. Dalej, dane te przedstawić można w dwóch wymiarach, gdzie na osi x oznacza się możliwe do uzyskania wyniki (zaokrąglone), a na osi y częstość z jaką model je uzyskiwał. Częstość łatwo zaś przekształcić można w prawdopodobieństwo[257], uzyskując stąd dyskretną[258] dystrybucję prawdopodobieństwa.

Przyjęto tradycyjnie, że istotne są wyniki, których prawdopodobieństwo wystąpienia p jest niższe niż 5% (*i.e*. próg istotności $\alpha = 0.05$).

W przypadku kryteriów: i) losowano 5000 podzbiorów o wielkości danego podzbioru kryterialnego; ii) zaokrąglano trafności do liczb naturalnych (101 możliwych trafności; od 0 do 100); iii) przedstawiano dystrybucję w dwu-wymiarowej przestrzeni (trafności na osi x, prawdopodobieństwo uzyskania tych trafności na osi y).

W przypadku kategorii: i) ogólna liczba wykonawców testowych wynosiła 145, możliwe więc było 145 rozmiarów podzbiorów wykonawców; ii) dla każdego możliwego rozmiaru losowano po 500 podzbiorów wykonawców o takiej wielkości; iii) określono 14645 prawdopodobieństw (101 trafności wobec 145 rozmiarów podzbiorów); iv) dystrybucję określono w trzy-wymiarowej przestrzeni (liczba wykonawców w podzbiorze na osi x, trafność na osi y, prawdopodobieństwo uzyskania trafności ze względu na wielkość podzbioru na osi z).

257 Dla każdego możliwego rezultatu, sprawdza się jak często wystąpił i dzieli się tą liczbę przez liczbę przeprowadzonych prób (*i.e.* przez liczbę przetestowanych podzbiorów).

258 Dyskretną (*i.e.* nie ciągłą), bowiem znane jest prawdopodobieństwo wystąpienia tylko określonych rezultatów, których częstość występowania policzono, a nie są znane prawdopodobieństwa wystąpienia rezultatów pośrednich pomiędzy tymi.

Terminologia. Posługiwano się następującymi miarami rezultatów: i) *Loss* (lub *Loss Function*), koszt dany funkcją kosztu; ii) Acc (*Accuracy*), trafność lub dokładność; iii) TPR (*True Positive Rate*), czułość lub wskaźnik prawdziwie pozytywnej klasyfikacji; iv) TNR (*True Negative Rate*), swoistość lub wskaźnik prawdziwie negatywnej klasyfikacji; v) FPR (*False Positive Rate*), błąd pierwszego rodzaju lub wskaźnik fałszywie pozytywnej klasyfikacji; vi) FNR (*False Negative Rate*), błąd drugiego rodzaju lub wskaźnik fałszywie negatywnej klasyfikacji; vii) PPV (*Positive Predictive Value*), pozytywna wartość predykcyjna; viii) NPV (*Negative Predictive Value*), negatywna wartość predykcyjna; ix) AUC (*Area Under the ROC Curve*), pole powierzchni pod krzywą ROC (*Receiver Operating Characteristic*)[259].

6.3. Rezultaty i dyskusja.

Metody oparte na podzbiorach.

Kryteria. Rozważając rezultaty kryteriów Podstawowego, IAM, CVL i CVL-Rozłącznego (tab. 6.3.1 i 6.3.2; rys. 6.3.1 i 6.3.2), więcej dysproporcji pomiędzy wynikami zachodzi dla modelu v2.4.0 niż v2.5.1. Ponadto, rezultaty dla kryteriów negatywnych są istotnie lepsze niż dla pozostałych kryteriów, pomimo że zastosowano: i) odszumianie i binaryzację danych w przypadku modelu v2.4.0; ii) wykluczenie par pomiędzy bazami danych w przypadku modelu v2.5.1. Podczas gdy rezultaty kryterium Negatywnego-Odszumionego są istotnie gorsze niż kryterium Negatwnego-Surowego dla modelu v2.5.1, to rezultaty kryterium Negatywnego dla modelu v2.4.0 są niemal stuprocentowe, stąd przypuszczać można, że szum nie jest koniecznym determinantem tych anomalii. Rezultaty kryterium Średniego dla modelu v2.4.0 są niższe niż rezultaty kryterium Podstawowego, ponieważ te ostatnie zawyżane były przez przypadki zachodzące pomiędzy bazami danych, które uwzględniono podczas treningu i testowania modelu v2.4.0. Podobnie, dla modelu v2.5.1 rezultaty kryterium Średniego są wyższe niż Podstawowego, uwzględnienie więc przypadków kryterium Negatywnego w teście podstawowym niesłusznie zawyżyłoby wyniki modelu.

259 Krzywa ROC obrazuje stosunek czułości do błędu pierwszego rodzaju dla różnych progów decyzyjnych klasyfikatora. Natomiast AUC określa jak dobrze model zdolny jest klasyfikować, niezależnie od wartości progów decyzyjnych.

Kryterium	Koszt	Trafność	TPR	TNR	FPR	FNR	PPV	NVP	AUC
Podstawowe	0.1661	**0.9481**	**0.9330**	**0.9632**	0.0368	0.0670	0.9621	0.9349	0.9859
IAM	0.1839	0.9420	0.9323	0.9517	0.0483	0.0677	0.9507	0.9336	0.9833
CVL	0.2446	0.9153	0.9336	0.8969	0.1031	0.0664	0.9006	0.9311	0.9689
CVL-Rozłączny	0.2429	0.9207	0.9389	0.9025	0.0975	0.0611	0.9060	0.9366	0.9706
Negatywne	0.0209	**0.9922**	Brak	0.9922	0.0078	Brak	Brak	Brak	Brak
Średnia	0.2142	**0.9286**	**0.9329**	**0.9243**	0.0757	0.0670	0.9256	0.9323	0.9761

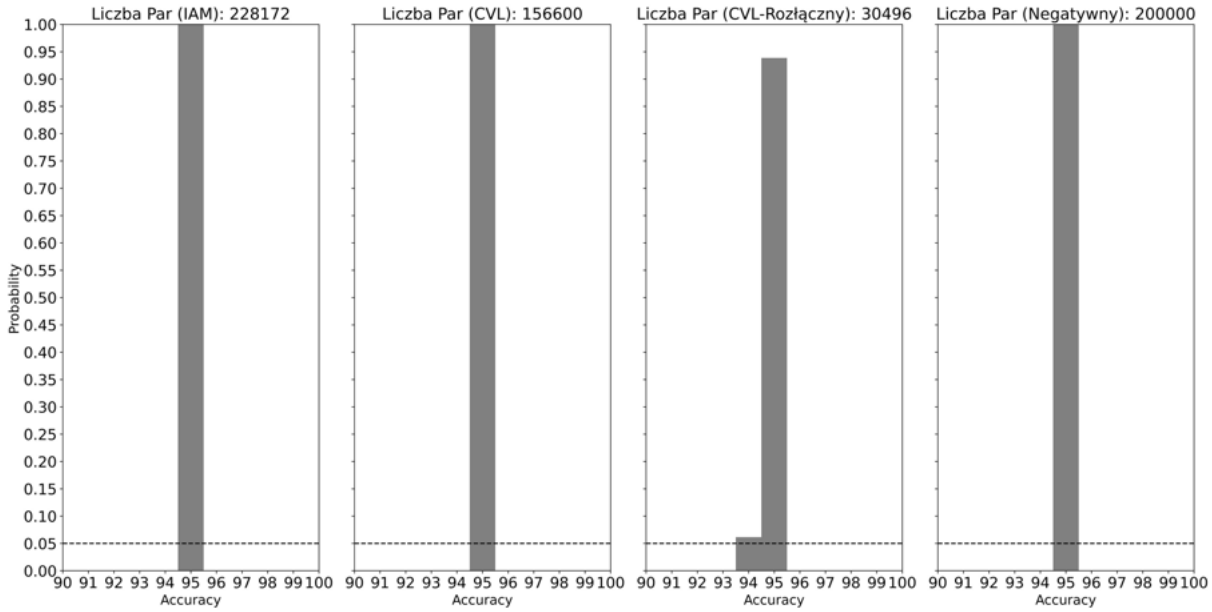
Tabela 6.3.1. Rezultaty ewaluacji modelu v2.4.0 na podstawie kryteriów.

Źródło: opracowanie własne.

Kryterium	Koszt	Trafność	TPR	TNR	FPR	FNR	PPV	NVP	AUC
Podstawowe	0.1300	**0.9582**	**0.9529**	**0.9636**	0.0364	0.0471	0.9632	0.9534	0.9891
IAM	0.1150	0.9661	0.9607	0.9714	0.0286	0.0393	0.9711	0.9611	0.9912
CVL	0.1628	0.9445	0.9430	0.9459	0.0541	0.0570	0.9458	0.9431	0.9846
CVL-Rozłączny	0.1284	0.9557	0.9654	0.9459	0.0541	0.0346	0.9469	0.9647	0.9892
Negatywne-Surowe	0.0264	**0.9897**	Brak	0.9897	0.0103	Brak	Brak	Brak	Brak
Negatywne-Odszumione	0.0818	0.9736	Brak	0.9736	0.0264	Brak	Brak	Brak	Brak
Średnia	0.1059	**0.9659**	Brak	0.9686	0.0314	Brak	Brak	Brak	Brak

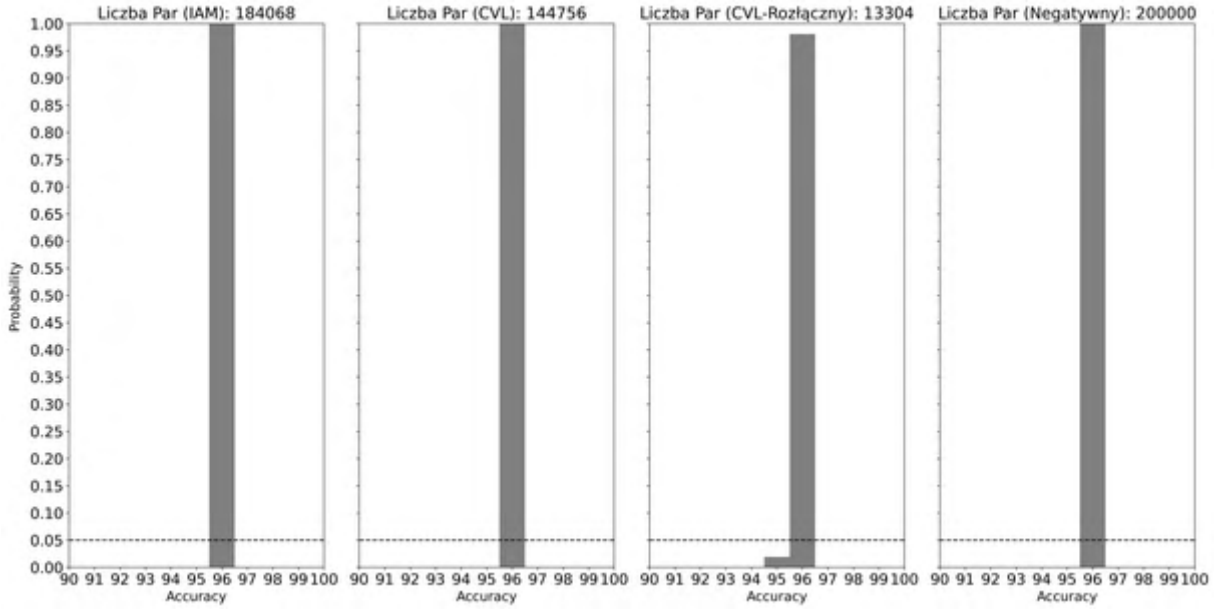
Tabela 6.3.2. Rezultaty ewaluacji modelu v2.5.1 na podstawie kryteriów.

Źródło: opracowanie własne.



Rysunek 6.3.1. Dyskretna dystrybucja prawdopodobieństwa uzyskania trafności ze względu na daną liczbę par obrazów dla modelu v2.4.0. Gdzie trafność (*Accuracy*) oznaczono na osi x, zaś prawdopodobieństwo (*Probability*) na osi y. Kryterium podstawowe, a więc cały zbiór testowy liczył 406548 par fragmentów.

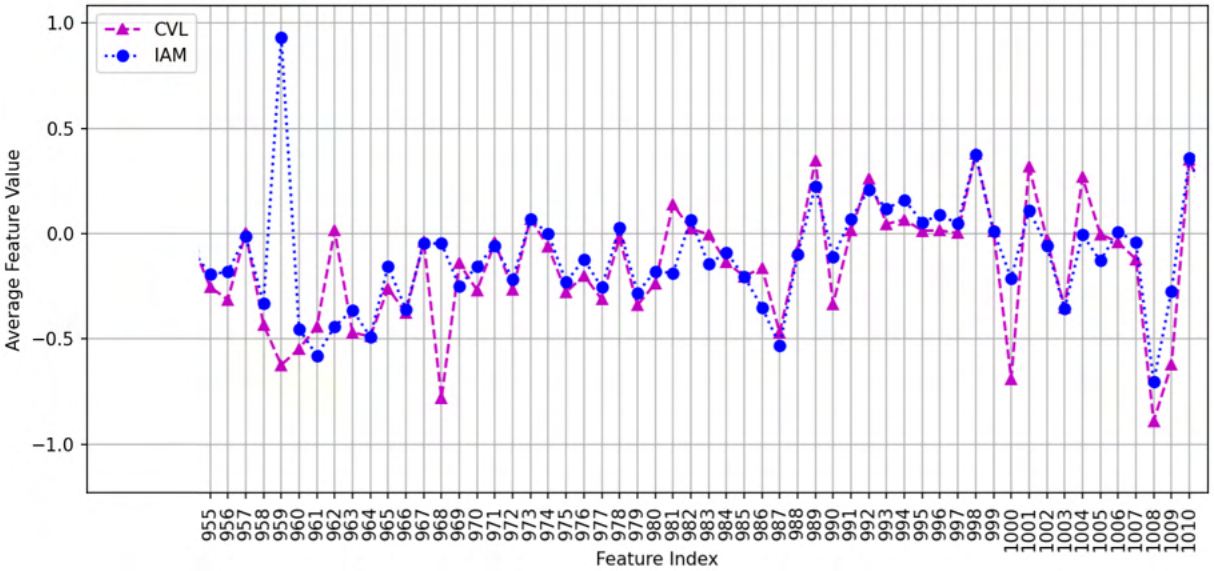
Źródło: opracowanie własne.



Rysunek 6.3.2. Dyskretna dystrybucja prawdopodobieństwa uzyskania trafności ze względu na daną liczbę par obrazów dla modelu v2.5.1. Gdzie trafność (*Accuracy*) oznaczono na osi x, zaś prawdopodobieństwo (*Probability*) na osi y. Kryterium podstawowe, a więc cały zbiór testowy liczył 328824 par fragmentów.

Źródło: opracowanie własne.

Względem rezultatów Negatywnych, założyć można hipotezę, że istnieje przynajmniej jedna taka cecha, która jest najbardziej dyskryminatywna wobec par obrazów zachodzących pomiędzy bazami danych, ale w innych przypadkach najbardziej dyskryminatywna nie jest. Obliczono więc średnie z wektorów cech dla bazy IAM i CVL, sprawdzając, które cechy zaobserwować można z większą częstotliwością dla jednego ze zbiorów (rys. 6.3.3). Cechy odstające pomiędzy zbiorami usuwano jeżeli różnica ich znormalizowanych wartości przekraczała 0.25. Ogółem, dla modelu v2.5.1 wyzerowano 89 spośród 1024 filtrów. Następnie poddano ten model powtórnej ewaluacji (tab. 6.3.3 i rys. 6.3.4), gdzie Negatywne rezultaty nie są już istotnie różne od Podstawowych (ale wystąpiła istotna dysproporcja wskaźników w kierunku klasy pozytywnej). Uległy też zmniejszeniu różnice pomiędzy rezultatami dla kryteriów CVL i CVL-Rozłącznego, oraz dla kryteriów Podstawowego i IAM. Skoro hipoteza została uprawdopodobniona, ponieważ zbliżono rezultaty Negatywne do Podstawowych, to przypuszczalnie może być to skuteczna metoda sanowania modeli.



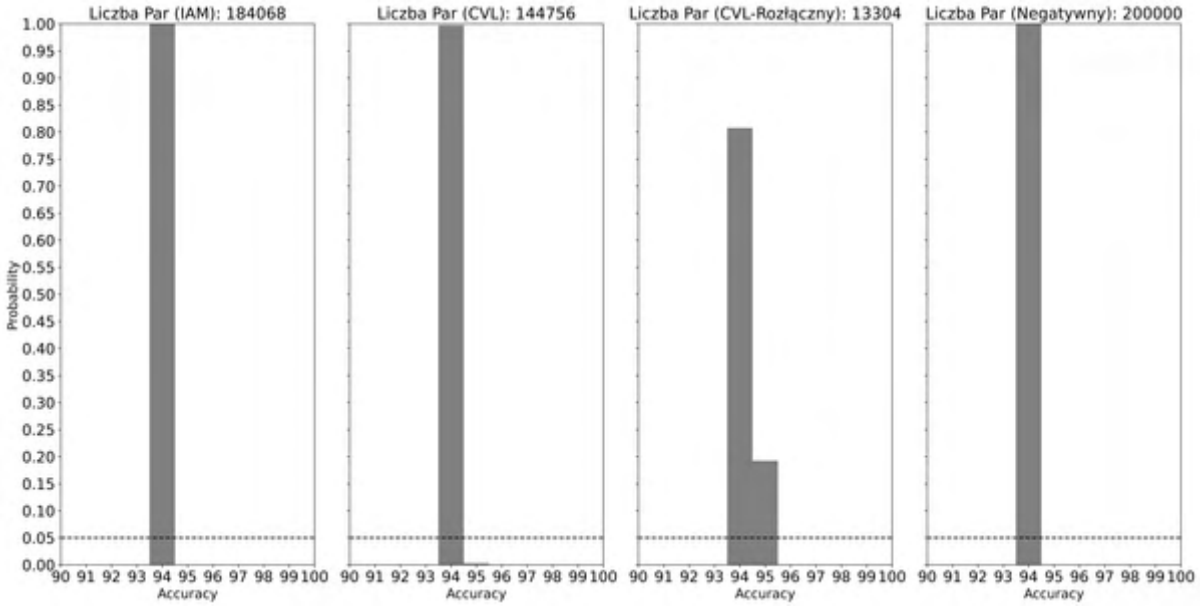
Rysunek 6.3.3. Przykładowy wycinek dystrybucji cech dla baz CVL i IAM. Na osi y oznaczono znormalizowaną średnią wartość cech (A*verage Feature Value*), zaś na osi x oznaczono identyfikatory cech (*de facto* filtrów odpowiedzialnych za ekstrakcję cech; *Feature Index*).

Źródło: opracowanie własne.

Kryterium	Koszt	Trafność	TPR	TNR	FPR	FNR	PPV	NVP	AUC
Podstawowe	0.1600	**0.9436**	**0.9831**	**0.9041**	0.0959	0.0169	0.9111	0.9817	0.9884
IAM	0.1400	0.9530	0.9861	0.9198	0.0802	0.0139	0.9248	0.9851	0.9907
CVL	0.2109	0.9248	0.9794	0.8703	0.1297	0.0206	0.8830	0.9769	0.9841
CVL-Rozłączny	0.1956	0.9292	0.9873	0.8712	0.1288	0.0127	0.8846	0.9856	0.9876
Negatywne-Surowe	0.1002	**0.9588**	Brak	0.9588	0.0412	Brak	Brak	Brak	Brak
Negatywne-Odszumione	0.1828	**0.9428**	Brak	0.9428	0.0572	Brak	Brak	Brak	Brak

Tabela 6.3.3. Rezultaty kryteriów dla modelu v2.5.1 po usunięciu odstających filtrów.

Źródło: opracowanie własne.



Rysunek 6.3.4. Dyskretna dystrybucja prawdopodobieństwa uzyskania trafności ze względu na daną liczbę par obrazów dla modelu v2.5.1 po usunięciu wybranych filtrów. Gdzie trafność (*Accuracy*) oznaczono na osi $\boldsymbol{x}$, zaś prawdopodobieństwo (*Probability*) na osi $\boldsymbol{y}$. Kryterium podstawowe, a więc cały zbiór testowy liczył 328824 par fragmentów.

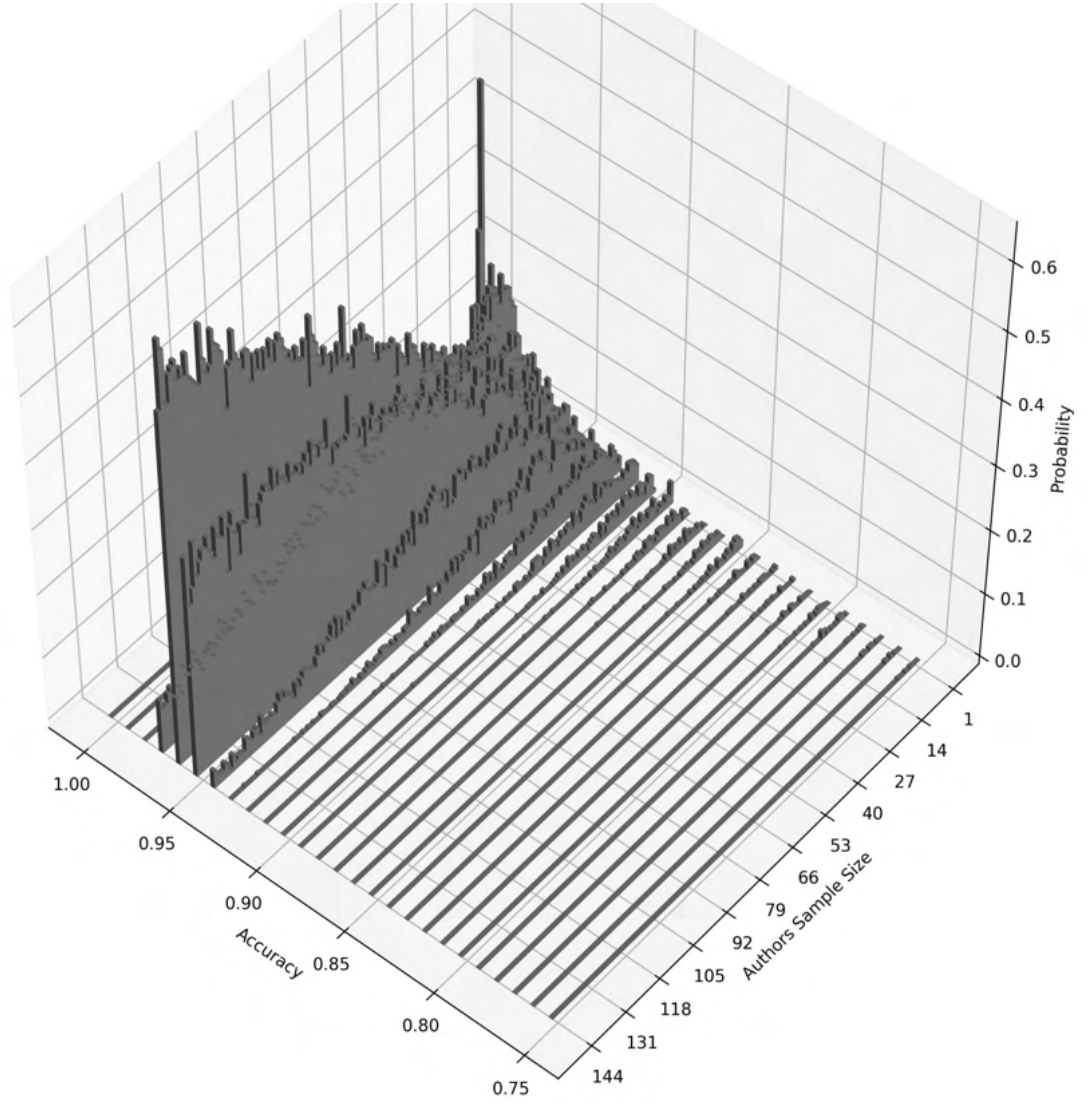
Źródło: opracowanie własne.

Kategorie. Rozważając rezultaty na kategoriach (tab. 6.3.4 i rys. 6.3.5), model v2.5.1 osiągnął istotnie niższe rezultaty dla najmniej licznego podzbioru, który oznaczono jako zbiór cech pisma kobiecego i leworęcznego. Najprawdopodobniej, kategoria ta różniła się od innych pod względem istotnych cech, a zarazem była podreprezentowana, stąd model nie nauczył się wystarczająco dobrze generalizować i odróżniać wykonawców należących do tej kategorii. Wnioskować stąd można, że model jest nierzetelny i nie powinien być stosowany w przypadku wykonawców, których cechy pisma przypisać można do wspomnianego zbioru (w rzeczywistości będzie dotyczyć to będzie zarówno kobiet, mężczyzn, leworęcznych i praworęcznych).

Liczba Wykonawców	Liczba Par	Kategoria Pisma	Koszt	Trafność	TPR	TNR	FPR	FNR	PPV	NPV	AUC
145	**328824**	**Brak**	**0.1319**	**0.9582**	**0.9529**	**0.9636**	**0.0364**	**0.0471**	**0.9632**	**0.9534**	**0.9891**
66	76754	Kobiece	0.1672	0.9487	0.9414	0.9560	0.0440	0.0586	0.9553	0.9422	0.9843
79	91042	Męskie	0.1094	0.9634	0.9655	0.9614	0.0386	0.0345	0.9616	0.9653	0.9918
23	31406	Leworęczne	0.1894	0.9383	0.9360	0.9405	0.0595	0.0640	0.9403	0.9363	0.9807
122	136390	Praworęczne	0.1267	0.9592	0.9587	0.9597	0.0403	0.0413	0.9597	0.9587	0.9895
9	**10094**	**Kobiece-Leworęczne**	**0.3308**	**0.9151**	**0.8419**	**0.9883**	**0.0117**	**0.1581**	**0.9863**	**0.8621**	**0.9717**
14	**21312**	Męskie-Leworęczne	0.1289	0.9485	0.9806	0.9165	0.0835	0.0194	0.9215	0.9792	0.9911
57	66660	Kobiece-Praworęczne	0.1517	0.9495	0.9565	0.9426	0.0574	0.0435	0.9434	0.9559	0.9862
65	69730	Męskie-Praworęczne	0.1272	0.9578	0.9608	0.9547	0.0453	0.0392	0.9550	0.9606	0.9895

Tabela 6.3.4. Rezultaty ewaluacji modelu v2.5.1 na podstawie kategorii.

Źródło: opracowanie własne.



Rysunek 6.3.5. Dyskretna dystrybucja prawdopodobieństwa trafności ze względu na wielkość podzbioru wykonawców. Gdzie na osi x oznaczono trafność (*Accuracy*), na osi y wielkość podzbioru wykonawców (*Authors' Sample Size*), zaś na osi z prawdopodobieństwo (*Probability*) uzyskania danej trafności dla danej wielkości podzbioru.

Źródło: opracowanie własne.

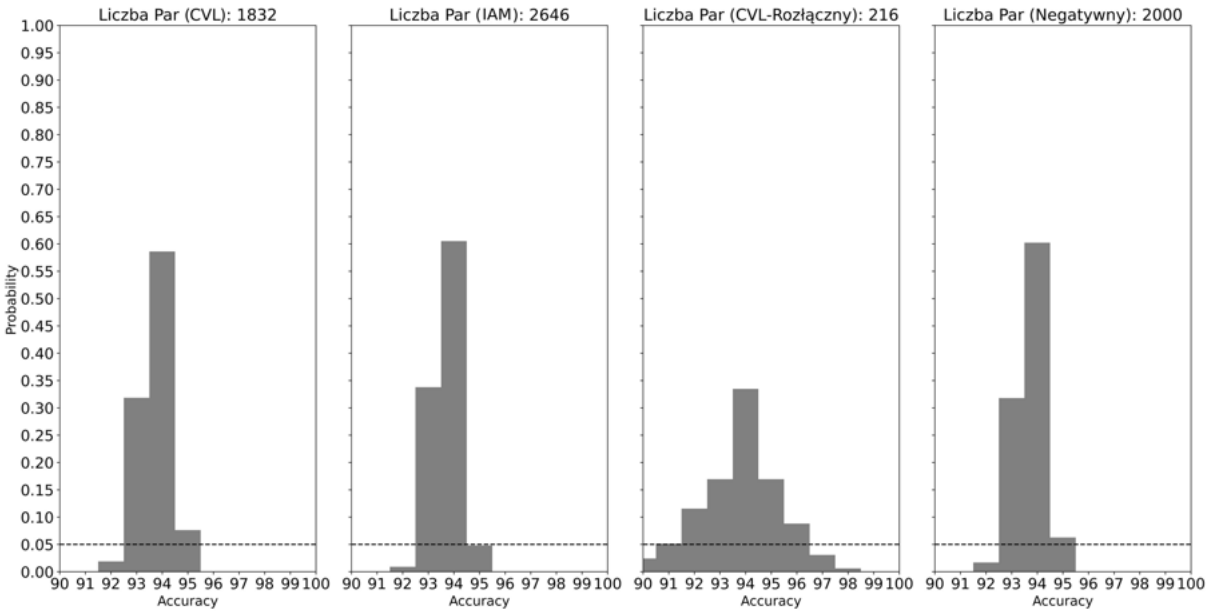
Metody oparte na danych wejściowych.

***Różne ilości danych*.** Można zaobserwować (tab. 6.3.2 i 6.3.5; rys. 6.3.2 i 6.3.6), że rezultaty dla ekstraktów dokumentów są istotnie różne pomiędzy sobą, są niższe niż dla fragmentów dokumentów (za wyjątkiem wyników Negatywnych), oraz że występuje istotna dysproporcja wskaźników w kierunku klasy pozytywnej.

Kryterium	Koszt	Trafność	TPR	TNR	FPR	FNR	PPV	NVP	AUC
Podstawowe	0.1700	**0.9352**	**0.9647**	**0.9058**	0.0942	0.0353	0.9110	0.9625	0.9838
IAM	0.1116	0.9580	0.9913	0.9247	0.0753	0.0087	0.9294	0.9906	0.9941
CVL	0.1972	0.9233	0.9463	0.9002	0.0998	0.0537	0.9046	0.9437	0.9768
CVL-Rozłączny	0.2243	**0.8935**	**0.9815**	**0.8056**	0.1944	0.0185	0.8346	0.9775	0.9851
Negatywne	0.0226	0.9925	Brak	0.9925	0.0075	Brak	Brak	Brak	Brak

Tabela 6.3.5. Rezultaty ewaluacji modelu v2.5.1 na podstawie ekstraktów.

Źródło: opracowanie własne.



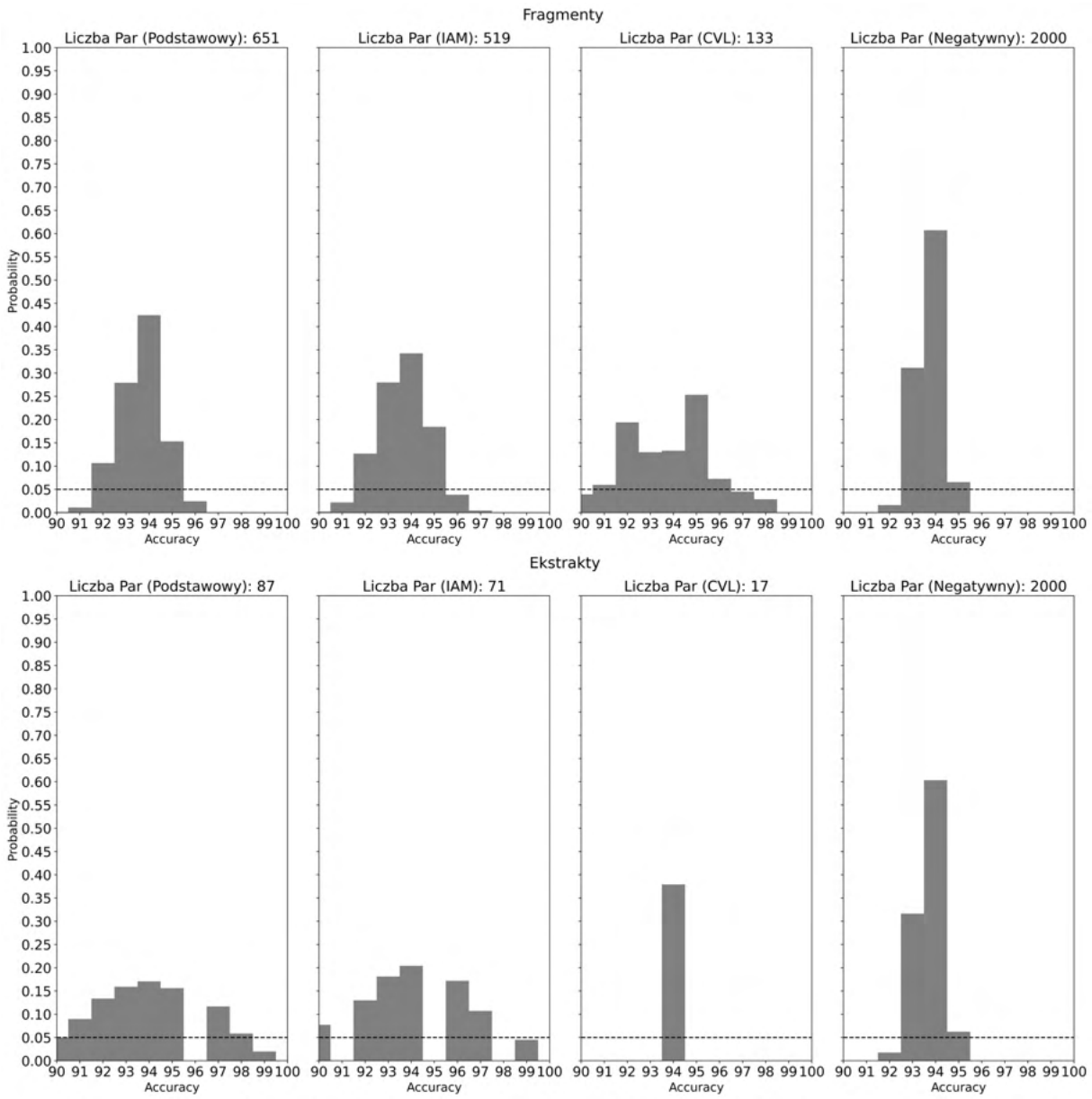
Rysunek 6.3.6. Dyskretna dystrybucja prawdopodobieństwa uzyskania trafności ze względu na daną liczbę par obrazów dla modelu v2.5.1 testowanego na ekstraktach dokumentów. Gdzie trafność (*Accuracy*) oznaczono na osi x, zaś prawdopodobieństwo (*Probability*) na osi y. Kryterium podstawowe, a więc cały zbiór testowy liczył 4478 par fragmentów.
Źródło: opracowanie własne.

Hipoteza autora jest taka, że dystrybucja cech na fragmentach i ekstraktach jest różna, stąd wynika różnica w rezultatach. Jeżeli zaś istnieje taki zbiór fragmentów i ekstraktów, gdzie każdy fragment posiada przynajmniej jeden ekwiwalentny pod względem wykonawstwa i dystrybucji cech ekstrakt, to rezultaty powinny być ekwiwalentne dla obydwu podzbiorów. Określono więc taki zbiór fragmentów i ekstraktów, gdzie ekwiwalencję wyznaczono poprzez dystans kosinusowy wektorów cech. Najlepsze rezultaty uzyskano, gdy: i) wobec wektorów cech zastosowano uprzednią normalizację L1, gdzie suma bezwzględnych wartości wektora wynosi 1; ii) maksymalny dystans kosinusowy określono jako 0.05. Ogółem, wyznaczono 84 pary ekwiwalentnych ekstraktów i fragmentów (24 ekstrakty i 60 fragmentów), na podstawie których przetestowano model v2.5.1 (kryterium CVL-Rozłączone nie było możliwe, ze względu na niewielką liczbę par). Jak można zaobserwować (tab. 6.3.6 i rys. 6.3.7), rezultaty ewaluacji na kryteriach są zbliżone i nie są przypadkowe dla obydwu podzbiorów[260]. Na podstawie z-testu (*z-test for the difference of two proportions*)[261]

260 Przyjęto granice skali trafności w przedziale [0.9, 1.0] który zawiera relewantny wycinek dystrybucji.
261 T.G. Dietterich, *Approximate Statistical Tests for Comparing Supervised Classification Learning Algorithms*, „Neural Computation" t. 10 (1998).

stwierdzić natomiast można, że rezultaty pomiędzy ekwiwalentnymi fragmentami i ekstraktami nie są istotnie różne (za wyjątkiem kryterium Negatywnego).

Niemniej, zakładano, że model powinien osiągnąć podobne lub lepsze rezultaty przy większej ilości danych, podczas gdy osiąga on rezultaty istotnie niższe, bo nie jest on wystarczająco odporny na nieznane mu dystrybucje cech pisma.



Rysunek 6.3.7. Dyskretna dystrybucja prawdopodobieństwa uzyskania trafności ze względu na daną liczbę par obrazów dla modelu v2.5.1 testowanego na ekstraktach i fragmentach dokumentów. Gdzie trafność (*Accuracy*) oznaczono na osi x, zaś prawdopodobieństwo (*Probability*) na osi y. Całość zbioru testowego liczyła 4478 par ekstraktów i 328824 par fragmentów.

Źródło: opracowanie własne.

Ilość Danych	Kryterium	Koszt	Trafność	TPR	TNR	FPR	FNR	PPV	NVP	AUC
Fragmenty	Podstawowe	0.0811	0.9716	0.9784	0.9648	0.0352	0.0216	0.9653	0.9781	0.9952
Ekstrakty	Podstawowe	0.1174	0.9651	1.0000	0.9302	0.0698	0.0000	0.9348	1.0000	0.9884
Fragmenty	IAM	0.0516	**0.9846**	**1.0000**	**0.9691**	**0.0309**	**0.0000**	**0.9700**	**1.0000**	**0.9961**
Ekstrakty	IAM	0.0302	**0.9857**	**1.0000**	**0.9714**	**0.0286**	**0.0000**	**0.9722**	**1.0000**	**1.0000**
Fragmenty	CVL	0.0098	0.9924	1.0000	0.9848	0.0152	0.0000	0.9851	1.0000	1.0000
Ekstrakty	CVL	0.0324	1.0000	1.0000	1.0000	0.0000	0.0000	1.0000	1.0000	1.0000
Fragmenty	Negatywne	0.0054	0.9965	Brak	0.9965	0.0035	Brak	Brak	Brak	Brak
Ekstrakty	Negatywne	0.0436	0.9715	Brak	0.9715	0.0285	Brak	Brak	Brak	Brak

Tabela 6.3.6. Rezultaty ewaluacji modelu v2.5.1 na podstawie ekwiwalentnych ekstraktów i fragmentów.

Źródło: opracowanie własne.

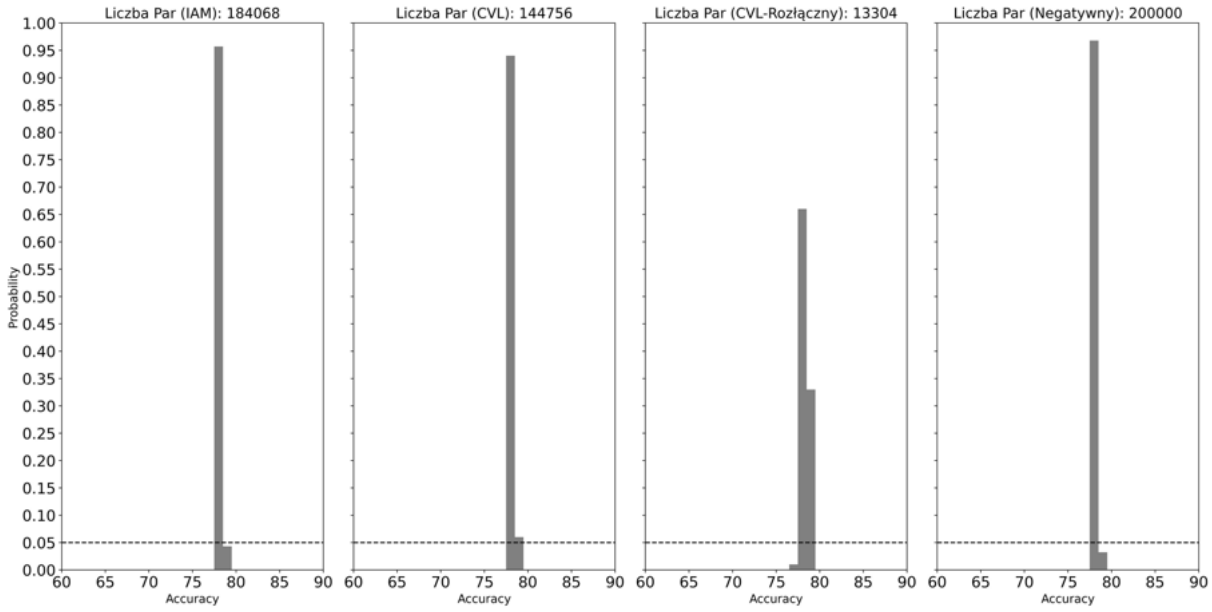
Różne jakości danych. Jak można zaobserwować (tab. 6.3.2 i 6.3.7; rys. 6.3.2 i 6.3.8), po naniesieniu na obrazy pisma kratki, rezultaty dla modelu v2.5.1 są istotnie niższe, oraz występuje istotna tendencja w kierunku klasy pozytywnej. Przypuszczać przy tym można, że podobny skutek miałoby naniesienie liniatury.

Poniższe rezultaty wynikają z nieprzystosowania modelu do przetwarzania takich obrazów, *i.e.* zbiory uczące nie zawierały takich przykładów. Nie jest przy tym istotne, że probanci pisali na papierze z podłożoną liniaturą. Nie byłoby też istotne, gdyby nie pisali na papierze z podłożoną liniaturą. Choć przypuszczać można, że istotne dysproporcje wystąpiłyby wtedy, kiedy zbiór treningowy nie zawierałby, a zbiór testowy zawierałby dokumenty pisane z podłożoną liniaturą (lub odwrotnie). Niemniej, najbardziej istotna jest tutaj obecność kratki na testowanych obrazach, a jej nieobecność w materiale uczącym. Najprostszym rozwiązaniem byłoby uwzględnienie w zbiorze uczącym prób pisma na papierze z liniaturą, kratką oraz na papierze czystym.

Kryterium	Koszt	Trafność	TPR	TNR	FPR	FNR	PPV	NVP	AUC
Podstawowe	1.2500	0.7833	0.8537	0.7129	0.2871	0.1463	0.7483	0.8297	0.8309
IAM	0.6368	0.8261	0.7773	0.8748	0.1252	0.2227	0.8613	0.7971	0.9001
CVL	2.7225	**0.6538**	**0.9509**	**0.3566**	0.6434	0.0491	0.5965	0.8790	0.7125
CVL-Rozłączny	2.5697	0.6539	0.9741	0.3337	0.6663	0.0259	0.5938	0.9281	0.7335
Negatywne	1.0418	0.7777	Brak	0.7777	0.2223	Brak	Brak	Brak	Brak

Tabela 6.3.7. Rezultaty ewaluacji modelu v2.5.1 na podstawie fragmentów z naniesioną kratką.

Źródło: opracowanie własne.



Rysunek 6.3.8. Dyskretna dystrybucja prawdopodobieństwa uzyskania trafności ze względu na daną liczbę par obrazów dla modelu v2.5.1 testowanego na fragmentach z naniesioną kratką. Gdzie trafność (*Accuracy*) oznaczono na osi x, zaś prawdopodobieństwo (*Probability*) na osi y. Kryterium podstawowe, a więc cały zbiór testowy liczył 328824 par fragmentów.

Źródło: opracowanie własne.

***Różna trudność danych*.** Wyniki testów na różnych poziomach trudności są w większości nieistotne (tab. 6.3.2 i 6.3.8; rys. 6.3.2 i 6.3.9, 6.3.10, 6.3.11). Rezultaty dla pierwszego poziomu trudności (test z wyłączeniem identycznych par fragmentów) wyjaśnić można relatywnie łatwo. Otóż identycznych przypadków było zaledwie 3384, więc brak ich miał niewielki wpływ na obniżenie wskaźników. Trudniej wyjaśnić jest wyniki dla drugiego poziomu trafności (test z wyłączeniem par fragmentów pochodzących z tego samego dokumentu), gdzie liczba par którą wyłączono ze zbioru testowego była dziesięciokrotnie większa (31996 przypadków, a więc 0.0973 liczby wszystkich przypadków testowych). Istotne okazały się rezultaty dla trzeciego poziomu trudności (test z wyłączeniem par fragmentów w tym samym języku), gdzie zaobserwowane różnice, nie są jednak adekwatne wobec wzrostu trudności testu.

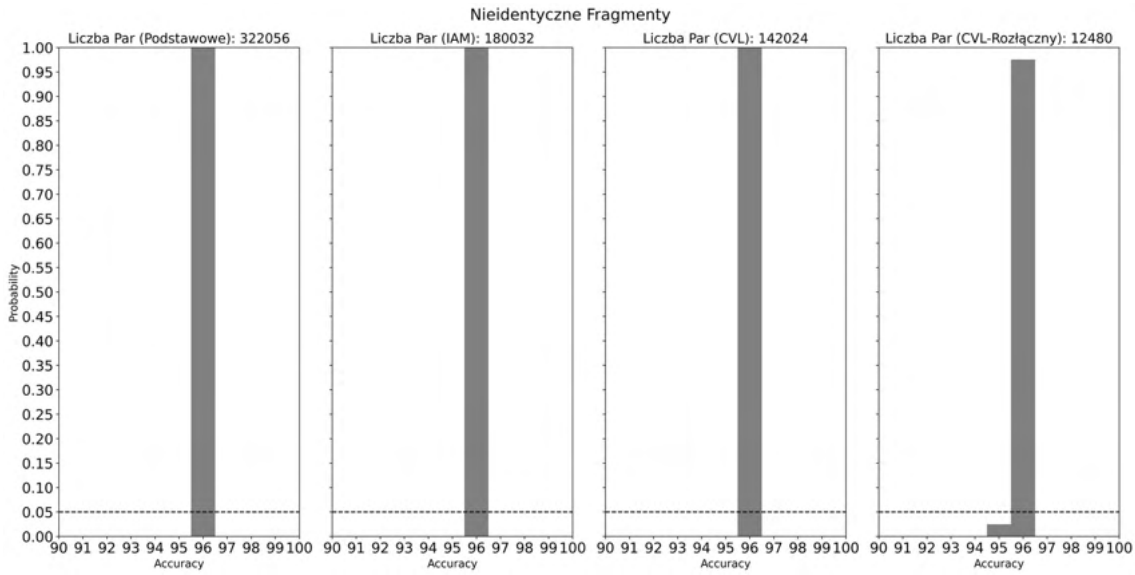
Jest to naturalne, że trudniejsze testy powinny prowadzić do niższych rezultatów, więc gdy różnice w wynikach są nieistotne pomimo trudniejszych testów, to może być tak, że różne poziomy trudności wcale nie są takimi dla modelu (nie są dla niego różnie trudne), bo nie rozważa on cech, których liczba i jakość zmniejsza się na kolejnych poziomach trudności. Jeżeli zmniejsza się liczba i jakość cech pismoznawczych, to model ich nie rozważa. Jeżeli są to zaś cechy inne niż

pismoznawcze, to nie możemy wykluczyć, że model rozważa cechy pismoznawcze. Trudno tutaj jednak rozstrzygnąć, który to z możliwych scenariuszy i w jakim stopniu. Podobnie, te różnice w wynikach, które są istotne, nadal nie przesądzają czy model rozważa cechy pismoznawcze. Istotą problemu jest tutaj kryterium podziału na stopnie trudności – dokąd może ono pociągać za sobą zmiany ilościowe i jakościowe cech innych niż pismoznawcze, dotąd nie będzie taki test definitywny, a skoro na ogół nie wiemy jakie cechy modele rozważają, to rezultaty zazwyczaj będą niedefinitywne.

Nieidentyczne	Kryterium	Koszt	Trafność	TPR	TNR	FPR	FNR	PPV	NVP	AUC
Fragmenty	Podstawowe	0.1300	**0.9579**	0.9519	0.9639	0.0361	0.0481	0.9635	0.9525	0.9891
	IAM	0.1164	0.9656	0.9598	0.9714	0.0286	0.0402	0.9711	0.9603	0.9910
	CVL	0.1640	0.9445	0.9419	0.9472	0.0528	0.0581	0.9469	0.9422	0.9844
	CVL-Rozłączny	0.1296	0.9560	0.9631	0.9489	0.0511	0.0369	0.9496	0.9626	0.9890
Dokumenty	Podstawowe	0.1445	**0.9550**	0.9468	0.9632	0.0368	0.0532	0.9626	0.9476	0.9877
	IAM	0.1310	0.9625	0.9548	0.9702	0.0298	0.0452	0.9697	0.9555	0.9893
	CVL	0.1709	0.9427	0.9374	0.9480	0.0520	0.0626	0.9475	0.9380	0.9834
	CVL-Rozłączny	0.1445	0.9511	0.9583	0.9440	0.0560	0.0417	0.9448	0.9577	0.9873
Języki	Podstawowe / CVL	0.1856	**0.9382**	0.9306	0.9458	0.0542	0.0694	0.9450	0.9317	0.9813
	CVL-Rozłączny	0.2186	0.9313	0.9188	0.9439	0.0561	0.0812	0.9424	0.9208	0.9766
	Negatywne	0.0265	0.9898	Brak	0.9898	0.0102	Brak	Brak	Brak	Brak

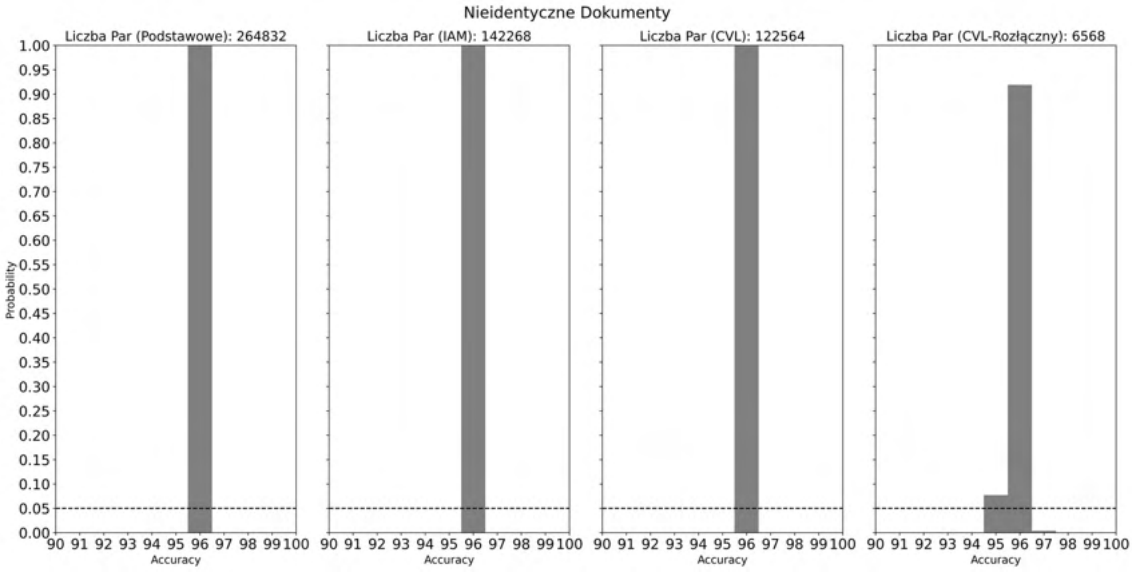
Tabela 6.3.8. Rezultaty ewaluacji modelu v2.5.1 z wyłączeniem identycznych par fragmentów, par fragmentów pochodzących z tego samego dokumentu i par fragmentów w tym samym języku.

Źródło: opracowanie własne.



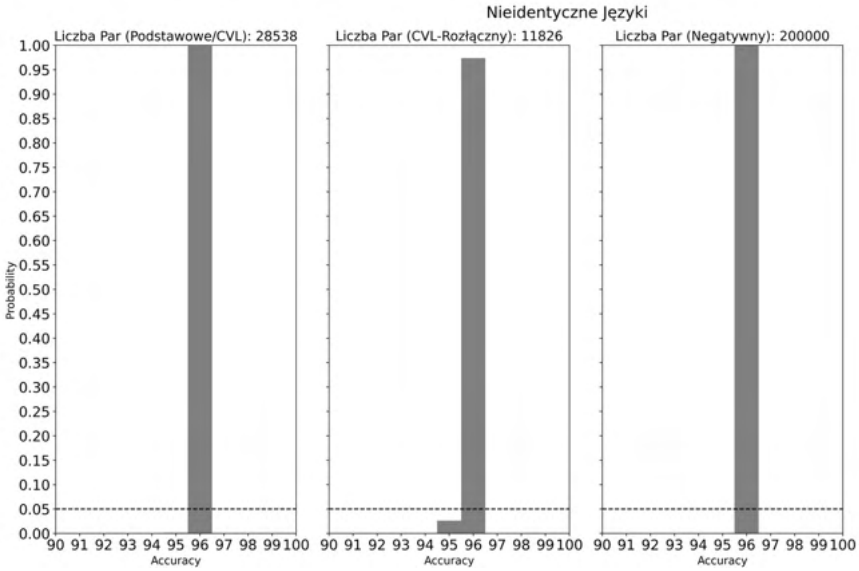
Rysunek 6.3.9. Dyskretna dystrybucja prawdopodobieństwa uzyskania trafności ze względu na daną liczbę par obrazów dla modelu v2.5.1 testowanego z wyłączeniem identycznych par fragmentów. Gdzie trafność (*Accuracy*) oznaczono na osi x, zaś prawdopodobieństwo (*Probability*) na osi y. Całość zbioru testowego liczyła 328824 par fragmentów.

Źródło: opracowanie własne.



Rysunek 6.3.10. Dyskretna dystrybucja prawdopodobieństwa uzyskania trafności ze względu na daną liczbę par obrazów dla modelu v2.5.1 testowanego z wyłączeniem fragmentów pochodzących z tego samego dokumentu. Gdzie trafność (*Accuracy*) oznaczono na osi x, zaś prawdopodobieństwo (*Probability*) na osi y. Całość zbioru testowego liczyła 328824 par fragmentów.

Źródło: opracowanie własne.



Rysunek 6.3.11. Dyskretna dystrybucja prawdopodobieństwa uzyskania trafności ze względu na daną liczbę par obrazów dla modelu v2.5.1 testowanego z wyłączeniem fragmentów w tych samym językach. Gdzie trafność (*Accuracy*) oznaczono na osi x, zaś prawdopodobieństwo (*Probability*) na osi y. Całość zbioru testowego liczyła 328824 par fragmentów.

Źródło: opracowanie własne.

***Różne metody preprocesowania*.** Jak można zaobserwować (tab. 6.3.9), rezultaty dla modelu v2.5.1 są istotnie pogorszone, wliczając nawet kryterium Negatywne, z powodu zupełnej stronniczości modelu w kierunku klasy pozytywnej.

Kryterium	Koszt	Trafność	TPR	TNR	FPR	FNR	PPV	NVP	AUC
Podstawowe	**20.2000**	**0.5064**	**0.9971**	**0.0156**	**0.9844**	0.0029	0.5032	0.8448	0.5111
IAM	16.2168	0.5092	0.9950	0.0235	0.9765	0.0050	0.5047	0.8232	0.5160
CVL	28.9528	0.5004	0.9999	0.0009	0.9991	0.0001	0.5002	0.8649	0.5007
CVL-Rozłączny	28.6659	0.5007	1.0000	0.0014	0.9986	0.0000	0.5004	1.0000	0.5013
Negatywne	**41.0964**	**0.0044**	Brak	**0.0044**	**0.9956**	Brak	Brak	Brak	Brak

Tabela 6.3.9. Rezultaty ewaluacji modelu v2.5.1 na podstawie binaryzowanych i odszumianych fragmentów.

Źródło: opracowanie własne.

Sprawdzając czy model rozróżnia wykonawców na podstawie cech narzędzi pisarskich, ustalono, że model w ogóle nie jest zdolny poradzić sobie z obrazami pisma uproszczonymi binaryzacją i odszumianiem. Problem polega na tym, że skala szarości umożliwia przechowanie i ekstrakcję wielu cech pismoznawczych, nie może być więc rozwiązaniem upraszczanie obrazów, a jednocześnie, nie można pozwolić aby modele dokonywały klasyfikacji w oparciu o cechy irrelewantne. Wnioskować więc można konieczność opracowania takiej bazy danych, gdzie probanci stosowali ten sam rodzaj narzędzia pisarskiego (jedyną bazą danych tego rodzaju, znaną autorowi, jest CEDAR-LETTER[262]). Jednakże, jeżeli stosować taki zbiór danych do uczenia modelu, zwłaszcza gdyby był to jedyny zbiór uczący, to spowodować można, że model będzie niezdatny do użytku w przypadku próbek sporządzonych odmiennymi narzędziami pisarskimi.

Ponadto, zauważyć należy na podstawie powyższych wyników, że zaistnieć może w praktyce taka sytuacja, gdzie konieczna będzie re-ewaluacja modelu przez eksperta. Na przykład, gdy dokument kwestionowany lub skaner był niskiej jakości, więc zastosowano jakąś szczególną technikę preprocesowania obrazu zanim wprowadzono go do modelu. Będzie koniecznym, aby dokonać powtórnej ewaluacji modelu, *e.g*. na jego danych testowych, ale preprocesowanych w taki sposób jak ten dokument kwestionowany.

262 S. Srihari, Y.-C. Shin, S. Lee, V. Govindaraju, S.-H. Cha, C.I. Tomai, B. Zhang, A. Shekhawat, D. Bartnik, W. Yang, S. Setlur, P. Kilinskas, F. Kunderman, X. Liu, Z. Shi, V. Ramanaprasad, *Method and apparatus for analyzing and/or comparing handwritten and/or biometric samples. United States Patent US7580551B1, filed 30 June 2003, and issued 25 August 2009* [na:] https://patents.google.com/patent/US7580551/en, dostęp 13 marca 2023 r.

Różne źródła danych. W przypadku modeli uczonych na obrazach ze zbioru CVL, model v1.1.0 sklasyfikował wszystkie pary ze zbioru IAM jako negatywne, podczas gdy model v2.1.0 sklasyfikował wszystkie pary ze zbioru IAM jako pozytywne (tab. 6.3.10). Przypuszczając, że wyniki te powodować może niższa jakość skanów cechująca obrazy z bazy IAM (znaczne zaszumienie), autor dokonał odszumiania obrazów ze zbioru IAM, poprzez zerowanie wartości pikseli niższych niż 55. W wyniku czego, wskaźniki klasowe częściowo odzyskały równowagę, ale pozostały istotnie niższe w porównaniu do CVL i CVL-Rozłączony.

Model	Kryterium	Koszt	Trafność	TPR	TNR	FPR	FNR	PPV	NVP	AUC
v1.1.0	CVL	0.2300	0.9124	0.9207	0.9040	0.0959	0.0792	0.9056	0.9194	0.9676
	CVL-Rozłączny	0.1795	0.9370	0.9617	0.9123	0.0876	0.0382	0.9164	0.9597	0.9781
	IAM	265.3726	**0.5047**	**0.0113**	**0.9979**	0.0020	0.9886	0.8478	0.5023	0.5058
	IAM-Odszumiony	0.3629	0.8835	0.9661	0.8009	0.1990	0.0338	0.8291	0.9594	0.9603
v2.1.0	CVL	0.1800	0.9387	0.9301	0.9474	0.0526	0.0699	0.9464	0.9312	0.9817
	CVL-Rozłączny	0.1494	0.9506	0.9567	0.9445	0.0555	0.0433	0.9452	0.9561	0.9858
	IAM	283.7893	**0.5266**	**0.9883**	**0.0649**	0.9350	0.0116	0.5138	0.8476	0.5269
	IAM-Odszumiony	0.6610	0.9138	0.9867	0.8408	0.1592	0.0133	0.8611	0.9845	0.9510

Tabela 6.3.10. Rezultaty ewaluacji modelu v1.1.0 i v2.1.0 na parach z bazy CVL i IAM.

Źródło: opracowanie własne.

Jest więc oczywistym, że modele powyższe nie stanowią rzetelnych metod, skoro nie radzą sobie z danymi analogicznymi do ich danych treningowych. Wnioskować stąd też można, że praktyczne zastosowania sieci neuronowych przez wymagać będą uprzedniej, a nawet każdorazowej ewaluacji modeli ze względu na daną grupę badanych i sprzęt, za pomocą którego digitalizowane będą próbki.

6.4. Wnioski. Jeżeli zostanie kiedyś osiągnięta interpretowalność procesów decyzyjnych sieci neuronowych, to umożliwi ona pozytywną ich ewaluację, *i.e.* zmierzającą do udowodnienia hipotezy o ich rzetelności. Zanim się to ziści, jedynie ewaluacja negatywna, zmierzająca do udowodnienia hipotezy o nierzetelności sieci neuronowych, będzie miarodajnym sposobem na ich ewaluację. Innymi słowy, najbardziej popularne a zarazem podstawowe testy, chociaż użyteczne, są niewystarczające. Niestety, nie trudno wskazać uniwersalny sposób na falsyfikację rzetelności sieci neuronowych, więc każdy przypadek musi być indywidualnie analizowany, celem oznaczenia adekwatnego zbioru hipotez falsyfikujących.

W celu zilustrowania potrzeby stworzenia metodyki ewaluacji sieci neuronowych, rozważony zostanie niesławny przypadek modelu, który nauczono odróżniać „kryminalistów” od „nie-kryminalistów” na podstawie zdjęć ich twarzy, a który osiągnął 90% trafność[263]. Z jednej strony, eugenika i fizjonomika zostały już dawno sfalsyfikowane i odrzucone, więc żadne pochodne tych hipotez nie mogą być prawdziwe i powinny zostać od razu odrzucone. Z drugiej strony, pomimo że szeroko rozprawiano na temat tego eksperymentu (przeprowadzonego w 2016 roku), to dopiero w roku 2020 zaproponowano hipotezę skutecznie falsyfikującą tamte rezultaty oraz rezultaty podobnych badań[264]. Otóż, zdjęcia „nie-kryminalistów” pozyskano za pomocą robota internetowego (*web-crawler*), a więc wszystkie pochodziły z różnych źródeł. Podczas gdy zdjęcia „kryminalistów” pochodziły w większości z jednej komendy policji, a więc z jednego źródła, a w pozostałym zakresie z listów gończych publikowanych przez miejscowe ministerstwo bezpieczeństwa publicznego, a więc z ograniczonej liczby źródeł. Autorzy eksperymentu byli świadomi tego problemu, ale założyli naiwnie, iż zniwelują go opierając się na fotografiach z dowodów osobistych, które wykonywane są przecież prywatnie, stąd nie powinny zawierać sygnatur cyfrowych charakterystycznych dla jednego źródła. Niestety, autorzy nie zwrócili uwagi na możliwość, że policja wykonuje zdjęcia lub skany fotografii z dowodów osobistych, a czyni to za pomocą ograniczonej liczby urządzeń, oraz że najprawdopodobniej poddaje je standardowej obróbce cyfrowej podczas wgrywania do swojej bazy danych. Przypuścić tutaj można najbardziej oczywiste metody ewaluacji, które należało zastosować w omawianym kontekście. Po pierwsze, model powinien był być przetestowany ze względu na obydwie bazy zdjęć „kryminalistów” z osobna (względem bazy zdjęć „nie-kryminalistów). Ponieważ zbiór zdjęć z listów gończych powinna cechować większa zmienność sygnatur cyfrowych (ze względu na większą ilość ich źródeł w porównaniu do zdjęć z komendy), to model powinien osiągać niższe rezultaty wobec bardziej niebezpiecznych przestępców. Po drugie, na świecie mniej jest przestępców wśród kobiet niż wśród mężczyzn, a przestępczość różni się pośród różnych kategorii wiekowych[265]. Różne odcienie skóry powodować mogą fałszywe

263 X. Wu, X. Zhang, *Responses to Critiques on Machine Learning of Criminality Perceptions (Addendum of arXiv:1611.04135)*, arXiv, 26 maja 2017 r., http://arxiv.org/abs/1611.04135.

264 K.W. Bowyer, M. King, W. Scheirer, K. Vangara, *The Criminality From Face Illusion*, arXiv, 18 listopada 2020 r., http://arxiv.org/abs/2006.03895.

265 D. Steffensmeier, *Age, Gender, and Crime Across Three Historical Periods: 1935, 1960, and 1985*, „Social Forces” t. 69 (1991).

predykcje modelu, jeżeli model ten nie był trenowany na zbiorze reprezentatywnym pod tym względem[266]. Jednakże, żadnych takich testów nie przeprowadzono, pomimo naukowej i etycznej obligacji, aby sprawdzić trafność modelu wobec połowy ludzkiej populacji, oraz jej podgrup wysoko wrażliwych (*vulnerable groups*), które często są podreprezentowane na zbiorach uczących. Przypuszczać można z dużą pewnością, że model nie zdałby ewaluacji względem powyższych kategorii statystycznych populacji ludzkiej. Po trzecie, autorzy eksperymentu sami dowiedli, że dodatek nawet niewielkiej ilości szumu istotnie obniżał wskaźniki trafności ich modelu, ale nie wyciągnęli wniosku, że sugeruje to niemożność zastosowania go w praktyce.

Ostatecznie, pamiętać należy, iż jedynym sposobem, aby dowieść, że sztuczna sieć neuronowa jest użytecznym i skutecznym narzędziem dla praktyki kryminalistycznej, jest podejmować jak najwięcej prób falsyfikacji i ponosić porażki.

6.5. Reprodukowalność. Wszystkie szczegóły, metody, rezultaty, dane i dodatkowe objaśnienia zostały udokumentowane i dostępne są w repozytorium autora na platformie Github: https://github.com/Ma-Marcinowski/Verificational-Model

Modele uczono i testowano za pomocą chmury obliczeniowej Google Colab(oratory), umożliwiającej korzystanie z GPU Nvidia Tesla K80. Gdzie przeciętna epoka uczenia modelu trwała 85 min (około 250 ms/krok; około 20552 kroków na epokę). Model v1.1.0 osiągnął najlepszy rezultat po 5 epokach, model v2.1.0 po 3 epokach, model v2.4.0 po 5 epokach i model v2.5.1 po 2.5 epokach.

266 J. Buolamwini, T. Gebru, *Gender Shades: Intersectional Accuracy Disparities in Commercial Gender Classification* [w:] *W: Proceedings of the 1st Conference on Fairness, Accountability and Transparency. Proceedings of Machine Learning Research, sine loco*, 2018.

Rozdział 7. Przykład interpretacji sztucznych sieci neuronowych na przykładzie badań pismoznawczych.

7.1. Wprowadzenie. Rozważając warunki pod którymi dopuszczalnym byłoby zastosowanie sztucznych sieci neuronowych w kryminalistyce, jednym z najważniejszych jest interpretowalność. Sztuczne sieci neuronowe są nieinterpretowalne (tzw. czarna skrzynka, *black-box*), ponieważ ich procesy decyzyjne są zbyt skomplikowane aby je zrozumieć (*i.e.* nie są semantycznie sensowne, ponieważ nie wiadomo jak je nazwać). Stąd, modele te są trudno falsyfikowalne[267], gdyż z nienazywalnych procesów decyzyjnych (*i.e.* z nieznanych zdań) nie da się wyprowadzić hipotez na ich temat. Tak więc, bez interpretowalności, falsyfikowalność sieci neuronowych jest wysoce ograniczona, a ich weryfikowalność niepewna. Po pierwsze, dlatego że sieci neuronowe mogą relatywnie łatwo osiągać wysokie wyniki nie ucząc się rozwiązywać problemu racjonalnie (*e.g.* zdolne są osiągać wysokie trafności, nawet kiedy przeklasyfikuje się ich dane uczące w sposób losowy)[268]. Po drugie, dlatego że interpretacja sieci neuronowych poprzez wizualizacje ich pobudzeń (względem danych wejściowych) jest wysoce ograniczonym narzędziem[269]. Ogółem, sieci neuronowe mogą być łatwo zweryfikowane (ewaluowane pozytywnie), ale bardzo trudno sfalsyfikowane (ewaluowane negatywnie). Jeżeli zaś nie da się wobec danej metody postawić hipotezy i poddać jej próbie falsyfikacji, to nie jest to metoda naukowa.

Jeżeli rozróżnimy interpretowalność na dwa poziomy, głęboką i powierzchowną. Gdzie interpretowalność głęboka rozumiana jest jako zupełna wiedza o procesach decyzyjnych sieci neuronowych (*i.e.* wyjaśnialność na poziomie ich procesów neuronalnych / na poziomie rozumowań maszynowych). Natomiast interpretowalność powierzchowna rozumiana jest jako wiedza o semantycznie sensownych procesach decyzyjnych, które eksternalizowała sieć neuronowa, a które bezpośrednio zdeterminowały jej rozstrzygnięcia (*i.e.* wyjaśnialność na poziomie współzależnych zadań wykonywanych przez sieć / poziomie rozumowań ludzkich). To, zaprojektować

267 K. Popper, *The Logic of Scientific Discovery*, London 2002, s. 57–73.

268 C. Zhang, S. Bengio, M. Hardt, B. Recht, O. Viwnyals, *Understanding Deep Learning (Still) Requires Rethinking Generalization*, „Communications of the ACM" t. 64 nr 3 (2021), DOI: 10.1145/3446776.

269 C. Rudin, *Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead*, „Nature Machine Intelligence" t. 1 nr 5 (2019), DOI: 10.1038/s42256-019-0048-x.

możemy sieć neuronową według czynności wykonywanych przez ekspertów danej dziedziny, szczególnie gdy są te czynności/zadania współzależne i wykonywane hierarchicznie dla uzyskania rozstrzygnięć danego problemu, tworząc w ten sposób model powierzchownie interpretowalny[270]. Istotą powyższej koncepcji nie jest wybielanie czarnych skrzynek, ponieważ ani to użyteczne, ani możliwe. Istotą tej koncepcji jest wykorzystanie ogromnych zdolności aproksymacyjnych czarnych skrzynek. Poprzez uczczenie ich przekształcania danych surowych w semantycznie sensowne informacje, które będą jedynymi i bezpośrednimi determinantami wyjściowych rozstrzygnięć (zamiast uczenia sieci neuronowych przekształcania danych surowych wprost na rozwiązania problemów).

Zważając, że: i) problem weryfikacji jest znacznie bardziej generalny i abstrakcyjny niż problem identyfikacji (w którym dla zbioru n wykonawców wyróżnić wystarczy n cech dyskryminatywnych); ii) to, modele weryfikacyjne są mniej interpretowalne niż identyfikacyjne; iii) oraz modele weryfikacyjne wymagają znacznych ilości danych aby dokonać generalizacji, podczas gdy modele identyfikacyjne zapamiętywać mogą nawyki dowolnej liczby wykonawców. Autor postanowił przeprowadzić eksperyment utworzenia modelu powierzchownie interpretowalnego na przykładzie modelu do identyfikacji wykonawców dokumentów (statyczne obrazy pisma), gdyby bowiem ten eksperyment się nie powiódł, to nie powiódł by się też bardziej kosztowny wariant z modelem weryfikacyjnym. W tym celu, autor: i) pozyskał skany statycznych obrazów pisma z bazy odręcznie pisanych dokumentów; ii) określił liczbę probantów zasadną z punktu widzenia badań pismoznawczych; iii) opisał ich nawyki pisarskie. Następnie wytrenował jedną z najlepszych architektur sieci neuronowych, celem: i) ekstrakcji cech pisma ze wspomnianych dokumentów (podług cech zidentyfikowanych przez autora); ii) identyfikacji wykonawców na wyłącznej podstawie tych cech.

W przypadku praktycznym założyć możemy, że ekspert, który ma za zadanie zidentyfikować wykonawcę dokumentu kwestionowanego spośród dużej liczby osób podejrzanych, przeprowadza następujące czynności: i) zbiera materiał porównawczy od osób podejrzanych o wykonawstwo; ii) identyfikuje cechy ich pisma i opisuje nawyki

270 Tematyka ta została też częściowo przedstawiona w artykule: M. Marcinowski, *Top interpretable neural network for handwriting identification*, „Journal of Forensic Sciences” t. 67 nr 3 (2022), DOI: 10.1111/1556-4029.14978.

pisarskie; iii) analizuje dane i identyfikuje najbardziej prawdopodobnego wykonawcę; iv) trenuje i testuje model w oparciu o materiał porównawczy i zidentyfikowane cechy; v) zleca modelowi identyfikację wykonawcy dokumentu kwestionowanego; vi) sprawdza czy cechy wyekstraktowane przez model zgodne są z cechami dokumentu kwestionowanego, które sam lub sama zidentyfikowała; vii) sprawdza czy cechy, które zaważyły na rozstrzygnięciu modelu, są cechami, które zdeterminowały jego lub jej rozstrzygnięcie. Jeżeli rozstrzygnięcia eksperta i maszyny są zgodne, to wzmacniają opinię, jeżeli zaś są rozbieżne, to mogą wskazywać na luki lub nieścisłości opinii. Ogólnie rzecz biorąc, nie jest istotne jakie to procesy neuronalne umożliwiły ekspertowi i modelowi zidentyfikować cechy pisma, istotne jest to, że cechy te są semantycznie sensowne i determinują identyfikację wykonawcy w sposób interpretowalny.

7.2. Metody.

Dane. Zbiór 189 odręcznie pisanych dokumentów wykonawstwa 27 osób (skany jednostronnych dokumentów A4, obrazy statyczne) pobrany został z bazy CVL (*Computer Vision Lab; Institute of Computer Aided Automation Vienna University of Technology*)[271]. Był to podzbiór testowy tej bazy, gdzie: i) wyznaczono 7 wzorców tekstu w języku angielskim i niemieckim (tab. 7.2.1); ii) 27 probantów wykonało po jednym dokumencie na każdy z 7 wzorców (wzorce nr 1–8; podzbiór testowy bazy); iv) formularze drukowano na czystych kartkach A4, pod którymi umieszczano kartki liniowane (1.5 cm interlinii); v) probantów proszono o skopiowanie wzorców swoim „codziennym pismem” i narzędziem pisarskim, oraz o zaprzestanie pisania w przypadku wyczerpania wyznaczonej przestrzeni pisarskiej; vi) dokumenty skanowano za pomocą Lexmark X652de w standardzie 300 dpi, zapisywano w 24 bitowej skali kolorów RGB, format TIFF. Pozyskane dokumenty autor podzielił ze względu na wzorzec ich treści, otóż: i) podzbiór treningowy liczący 135 skanów zawierał dokumenty powstałe podług wzorców nr 1–6; ii) podzbiór testowy liczący 54 skany zawierał dokumenty powstałe podług wzorców nr 7 i 8.

271 F. Kleber, S. Fiel, M. Diem, R. Sablatnig, *CVL-DataBase: An Off-Line Database for Writer Retrieval, Writer Identification and Word Spotting* [w:] *2013 12th International Conference on Document Analysis and Recognition*, Washington, DC, USA 2013.

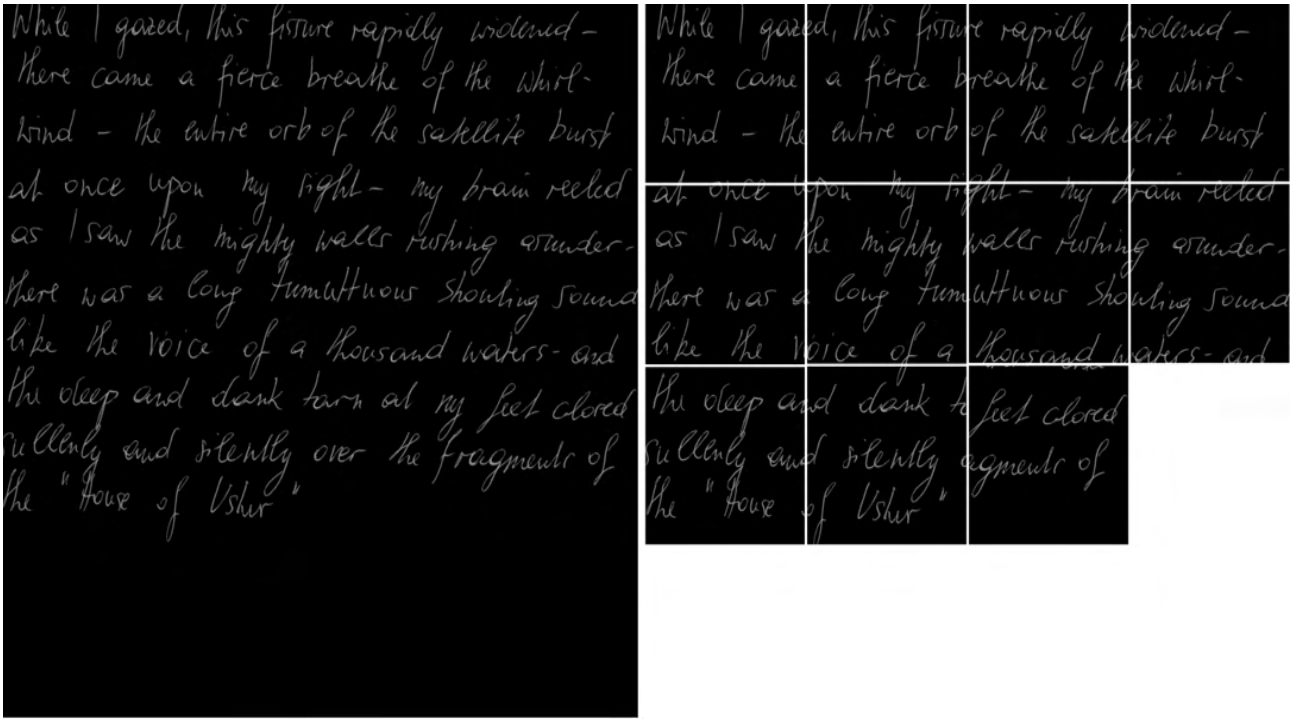
Identyfikator	Autor	Tytuł	Liczba słów
1	Edwin A. Abbot-Flatland	A Romance of Many Dimension	90
2	William Shakespeare	Mac Beth	47
3	**Wikipedia**	**Mailüfterl**	**74**
4	Charles Darwin	Origin of Species	52
6	**Johann Wolfgang von Goethe**	**Faust. Eine Tragödie**	**50**
7	Oscar Wilde	The Picture of Dorian Gray	65
8	Edgar Allan Poe	The Fall of the House of Usher	73

Tabela 7.2.1. Wzorce tekstów z bazy CVL (wytłuszczeniem oznaczono teksty w języku niemieckim). Źródło: F. Kleber, S. Fiel, M. Diem, R. Sablatnig, *CVL-DataBase: An Off-Line Database for Writer Retrieval, Writer Identification and Word Spotting* [w:] *2013 12th International Conference on Document Analysis and Recognition*, Washington, DC, USA 2013, s. 560–561.

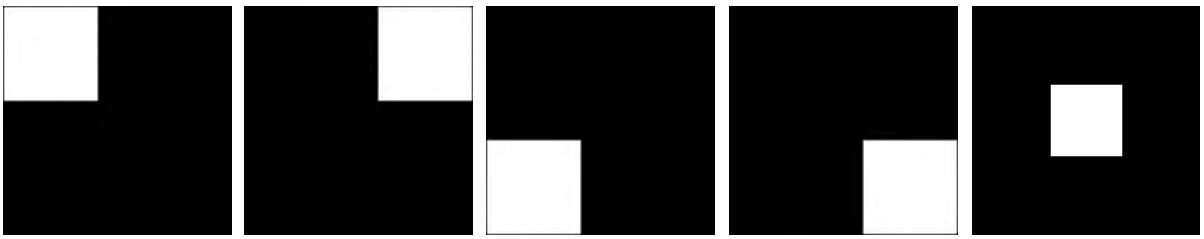
Preprocesowanie. Obrazy dokumentów przekształcane były do skali szarości, następnie przeprowadzano inwersję kolorów, ekstrakcję przestrzeni pisarskiej o wymiarach 2048 x 2048 px, redukcję rozmiaru ekstraktów do wymiarów 1024 x 1024 px, podział ekstraktów na fragmenty o wymiarach 256 x 256 px (rys. 7.2.1), konwersję z formatu TIFF do PNG.

Fragmenty, które nie zawierały tekstu lub zawierały małe jego ilości były odrzucane. Otóż, skoro dokonywano uprzedniej inwersji kolorów, to kolor tła był czarny zamiast białego, takie zaś piksele posiadają wartość zero, a suma pikseli na obrazie pozbawionym tekstu również wynosi zero. Stąd, na potrzeby selekcji fragmentów: i) odszumiano fragmenty poprzez progowanie wartości pikseli niższych niż 55 do zera; ii) wykonywano iloczyn fragmentu z czarno-białym filtrem o tym samym rozmiarze i sumowano uzyskane wartości; iii) jeżeli wynik sumy był dla wszystkich filtrów większy od zera to fragmentu nie pomijano. Filtry opracowano aby zapewnić wykorzystanie takich tylko fragmentów obrazów, gdzie tekst zachodził na wszystkie białe pola (rys. 7.2.2). Odnotować należy, że odszumianie fragmentów podczas ich filtrowania wykorzystywano tylko w tym celu.

Ogółem, pozyskano stąd 1366 fragmentów (412 testowych i 954 treningowe).



Rysunek 7.2.1. Przykładowy ekstrakt (po lewej) i jego fragmenty (po prawej) po preprocesowaniu.
Źródło: opracowanie własne.



Rysunek 7.2.2. Filtry zastosowane do selekcji fragmentów obrazów.
Źródło: opracowanie własne.

Cechy pisma. Autor przeprowadził opis nawyków pisarskich wszystkich wykonawców, uwzględniając ogółem 25 kategorii cech pisma, które opisał wieloetykietowo-binarnie (*i.e.* jako wiele rozłącznych problemów binarnych). Gdzie: i) jeżeli zamiarem było stwierdzenie występowania cech danej kategorii u danego wykonawcy, to kategoria taka ujęta była jako pojedyncza etykieta/problem (*e.g.* manieryzmy i ozdoby, które albo są albo ich nie ma); ii) jeżeli zamiarem było stwierdzenie jakie cechy danej kategorii występują u danego wykonawcy, to kategoria taka ujęta była jako wiele etykiet/problemów (*e.g.* typy pisma, gdzie każdy typ pisma jest lub go nie ma). Na przykład, definiując nachylenie pisma w czterech zakresach

(lewoskośne, proste, prawoskośne, głęboko prawoskośne), każdy zakres jest osobną etykietą (problemem do rozwiązania / cechą do ekstrakcji) i przyjmuje wartość prawdy lub fałszu (stąd opis wieloetykietowy-binarny). Zależnie od kategorii cech i sposobu jej etykietowania, niektóre etykiety wzajemnie się wykluczały (*e.g.* poziomy wyrobienia pisma), więc nie mogły być równocześnie prawdziwe. Inne etykiety się krzyżowały (*e.g.* zakresy nachylenia pisma), więc mogły być równocześnie prawdziwe. Ogółem wyznaczono 80 etykiet (rys. 7.3.1 i tab. 7.3.5) oraz cztery dodatkowe kategorie pisma (kobiecość, męskość, leworęczność i praworęczność pisma), stanowiące raczej zbiory cech lub cechy wyższego rzędu, a nie determinanty płci lub ręki dominującej.

Podobnie jak w przypadku wcześniej opisanych badań (rozdz. 6), dobór cech i sposób ich etykietowania zdeterminowany był przez system cech pisma zaproponowany przez Hubera *et al*.[272] oraz uczestników Jesiennej Szkoły Empirycznych Badań Pisma Ręcznego[273]. Ogółem, były to: i) wyrobienie, staranność, czytelność i dojrzałość; ii) typ pisma; iii) ogólny obraz pisma; iv) budowa wiązań międzyliterowych; v) impuls dominujący; vi) układ wierszy względem siebie; vii) kształt linii wierszy; viii) kierunek linii wierszy; ix) nachylenie pisma; x) zmienność nachylenia pisma (*quality of the slant*); xi) wielkość pisma; xii) odstępy między wierszami; xiii) odstępy między wyrazami; xiv) odstępy między znakami; xv) proporcje wysokości elementów nadlinijnych do wysokości elementów śródlinijnych; xvi) proporcje wysokości elementów podlinijnych do wysokości elementów śródlinijnych; xvii) kierunek kreślenia elementów poziomych; xviii) kierunek kreślenia elementów pionowych; xix) kierunek kreślenia owali, łuków i pętlic; xx) kierunek kreślenia inicjacji; xxi) kierunek kreślenia terminacji; xxii) siła nacisku i cieniowania; xxiii) kierunek nacisku i cieniowania; xxiv) znaki diakrytyczne; xxv) manieryzmy i ozdoby.

Na zbiór cech pisma kobiecego składały się: i) wysokie wyrobienie, staranność, czytelność i dojrzałość pisma; ii) ogólny obraz pisma owalny lub okrągły; iii) jednolite nachylenie pisma; iv) manieryzmy. Na zbiór cech pisma męskiego składały się: i) niskie wyrobienie, staranność, czytelność i dojrzałość pisma; ii) pismo kątowe; iii) nachylenie głębsze niż 70 stopni; iv) kreskowe znaki diakrytyczne nad „i”; v) duża siła nacisku lub cieniowania; vi) pismo drobne, duże lub bardzo duże. Na zbiór cech pisma

272 R.A. Huber, A.M. Headrick, H.H. Harralson, L.S. Miler, *Handwriting Identification Facts and Fundamentals*, Boca Raton 2018.

273 Instytut Ekspertyz Sądowych w Krakowie, *Słownik Terminów Pismoznawczych* [na:] http://prawouam-stp.home.amu.edu.pl/, 2007 r., dostęp 20 września 2021 r.

leworęcznego składały się: i) elementy poziome kreślone w kierunku wstecznym; ii) owale, łuki i pętlice kreślone zgodnie z ruchem wskazówek zegara; iii) pismo pionowe lub lewoskośne; iv) pismo o zmiennym nachyleniu; v) terminacje do góry i w lewo; vi) nacisk wstępujący. Podczas gdy wskaźnikiem przynależności do zbioru cech pisma praworęcznego był brak silnych wskazań przynależności do zbioru cech pisma leworęcznego.

Model. Architektura powierzchownie interpretowanej sieci neuronowej (*top interpretable neural network*, TINN) oparta została na jednej z najlepszych obecnie (*state of the art*) sieci neuronowych opracowanej przez Simonyana i Zissermana, VGG16[274]. Wprowadzono następujące modyfikacje tej architektury: i) pomiędzy warstwami konwolucyjnymi, oraz w pełni połączonymi, wprowadzono warstwy normalizujące (*batch-normalization*); ii) przed wszystkimi warstwami w pełni połączonymi zastosowano warstwy opuszczające; iii) na warstwach konwolucyjnych zastosowano dylatacje o wartości 2; iv) zamiast ostatniej warstwy redukującej poprzez wyciągnięcie najwyższej wartości, zastosowano warstwę wyciągającą średnią globalną; v) druga/przedostatnia warstwa w pełni połączona zawierała 84 neurony z sigmoidalną funkcją aktywacji (stosownie do liczby cech do wyekstraktowania); trzecia/ostatnia warstwa w pełni połączona liczyła 27 neuronów z funkcją aktywacji softmax (stosownie do liczby wykonawców do zidentyfikowania).

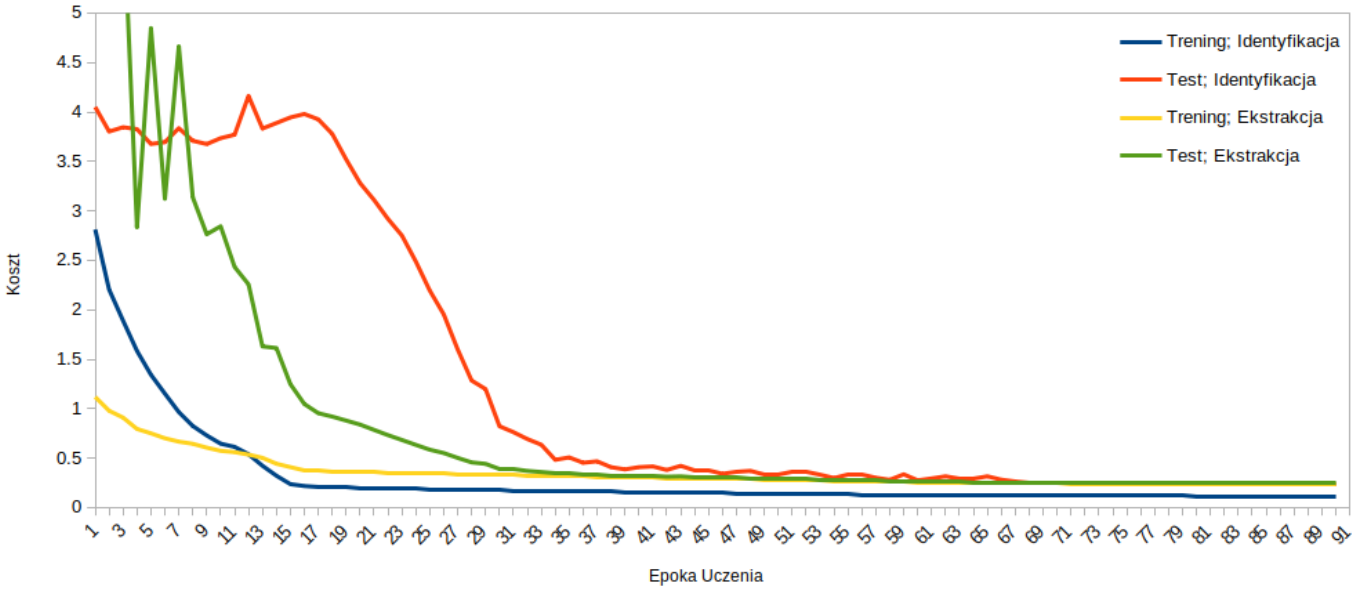
Model rozwiązywać miał dwa problemy hierarchicznie. W pierwszym kroku, model miał dokonać ekstrakcji cech pisma (problem wieloetykietowy-binarny), gdzie stosowano binarną trafność i koszt (binarna entropia krzyżowa, *binary cross-entropy*[275]). W drugim kroku, model miał za zadanie dokonać identyfikacji wykonawców na podstawie cech wyekstraktowanych w pierwszym kroku (problem jednoetykietowy-wieloklasowy), gdzie stosowano kategorialną trafność i koszt (kategorialna entropia krzyżowa, *categorical cross-entropy*[276]). Tak więc, na wyjściu z warstwy przedostatniej oczekiwano wektora 84 elementowego (bo tyle było etykiet nawiązujących do cech i zbiorów cech pisma), gdzie wartość każdej etykiety opisana była jako zero lub jeden (bo wiele etykiet to wiele cech do ekstrakcji, a dwie klasy to prawda i fałsz). Natomiast, na

274 K. Simonyan, A. Zisserman, *Very Deep Convolutional Networks for Large-Scale Image Recognition*, 10 kwietnia 2015 r., http://arxiv.org/abs/1409.1556.
275 A. Glassner, *Deep Learning: From Basics to Practice*, Seattle 2018, s. 1164–1165.
276 Ibid.

wyjściu z ostatniej warstwy oczekiwano wektora 27 elementowego (bo tylu było możliwych wykonawców), składającego się z 26 zer i 1 jedynki, która odpowiadać powinna danemu zidentyfikowanemu wykonawcy (gdzie jedna etykieta to jeden problem, a wiele klas to wielu wykonawców). Skąd właśnie, wskaźniki (*e.g*. trafność) i funkcje kosztu musiały być kompatybilne z rodzajami przedstawionych problemów. Aby sprawdzić podobieństwo wskaźników klasowych (TPR[277], TNR[278], FPR[279], FNR[280]), mierzono także AUC (*Area Under the ROC Curve*), czyli pole powierzchni pod krzywą ROC (*Receiver Operating Characteristic*), która obrazuje stosunek czułości do błędu pierwszego rodzaju wobec różnych progów decyzyjnych klasyfikatora, gdzie AUC określa jak dobrze model zdolny jest klasyfikować, niezależnie od wartości progów decyzyjnych. Było to szczególnie istotne w przypadku trafności binarnej, skoro bowiem średnia wartość wektorów cech określonych przez autora wynosiła 0.3496, to model który klasyfikowałby wszystkie cechy jako fałszywe, mógłby osiągnąć trafność 0.6504.

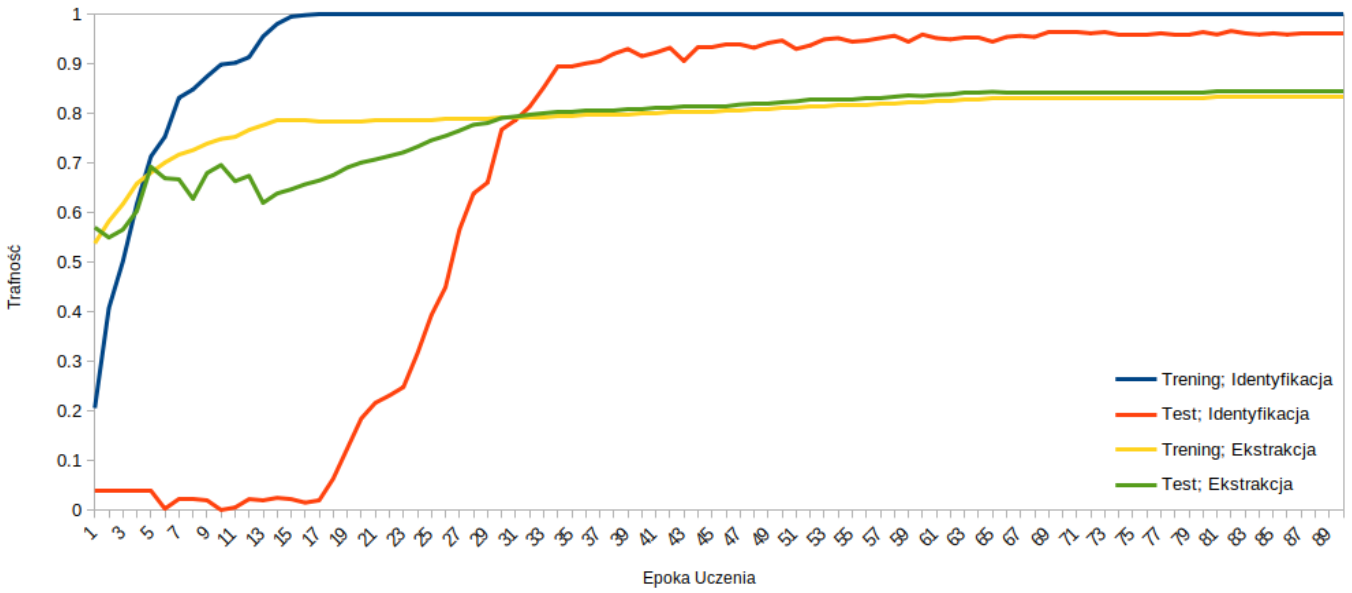
Pod względem parametrów, zastosowano: i) optymizator Adam (*Adaptive Moment Estimation*) z rekomendowanymi parametrami[281]; ii) automatyczne redukcje współczynnika uczenia poprzez współczynnik redukcji wynoszący 0.1 (wykonywano iloczyn współczynnika uczenia i redukcji, jeżeli koszt testowy nie poprawiał się przez pięć epok uczenia z rzędu); iii) *batch-size* wynosił 32; iv) prawdopodobieństwo zablokowania sygnału na warstwie opuszczającej wynosiło 0.0; v) trening trwał 90 epok (rys. 7.2.3 i 7.2.4).

277 *True Positive Rate*, czułość lub wskaźnik prawdziwie pozytywnej klasyfikacji.
278 *True Negative Rate*, swoistość lub wskaźnik prawdziwie negatywnej klasyfikacji.
279 *False Positive Rate*, błąd pierwszego rodzaju lub wskaźnik fałszywie pozytywnej klasyfikacji.
280 *False Negative Rate*, błąd drugiego rodzaju lub wskaźnik fałszywie negatywnej klasyfikacji.
281 D.P. Kingma, J. Ba, *Adam: A Method for Stochastic Optimization*, arXiv, 29 stycznia 2017 r., http://arxiv.org/abs/1412.6980.



Rysunek 7.2.3. Wykres funkcji kosztu podczas uczenia modelu TINN.

Źródło: opracowanie własne.



Rysunek 7.2.4. Wykres funkcji trafności podczas uczenia modelu TINN.

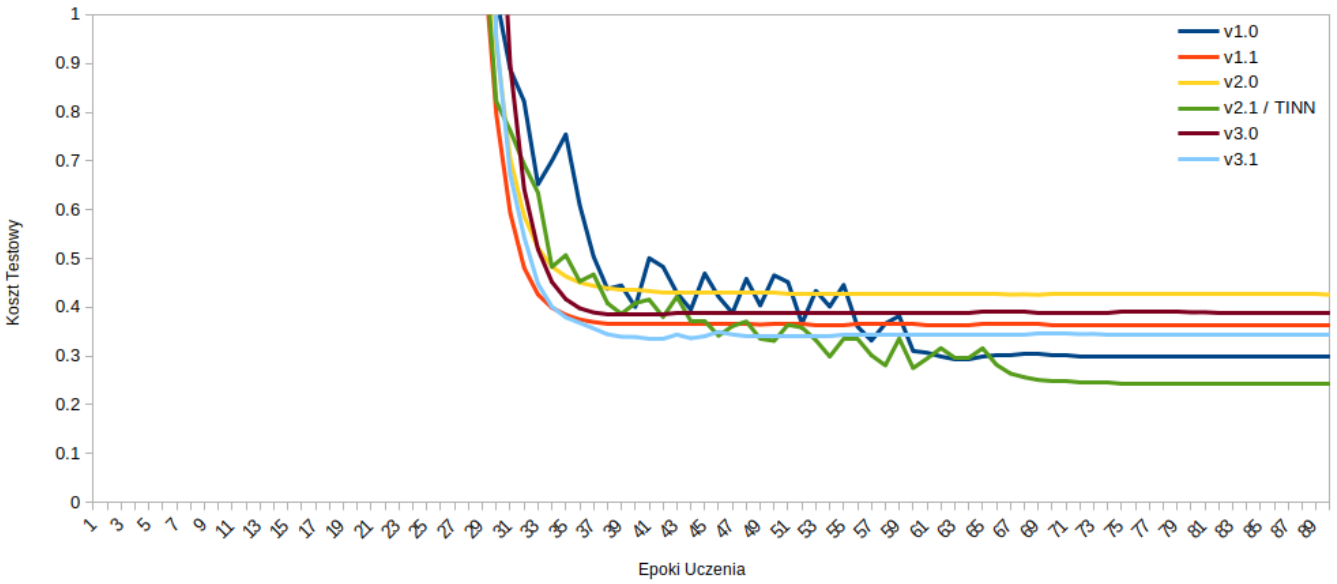
Źródło: opracowanie własne.

Modele porównawcze. W celu uzyskania porównywalnych rezultatów opracowano, wytrenowano i przetestowano następujące modele: v1.0) model analogiczny pod względem architektury do modelu TINN, ale jego zadaniem była wyłącznie identyfikacja wykonawców, stąd druga warstwa w pełni połączona nie została zmodyfikowana względem VGG16 i liczyła 4096 neuronów z funkcją aktywacji

ReLU; v1.1) model identyczny pod względem architektury do modelu TINN, ale jego wyłącznym zadaniem była identyfikacja wykonawców (stanowi on bezpośrednie porównanie do modelu TINN); v2.0) model identyczny pod względem architektury do v1.0, ale 84 spośród 4096 neuronów jego drugiej warstwy w pełni połączonej dokonywało ekstrakcji cech i było aktywowane funkcją sigmoidalną (*ergo* przepływ informacji do warstwy ostatniej nie był ograniczony do ekstrakcji cech narzuconych przez autora); v2.1) model TINN (*top interpretable neural network*, powierzchownie interpretowalna sieć neuronowa); v3.0) model identyczny w architekturze do modelu v1.0, ale jego zadaniem była równoległa ekstrakcja cech i identyfikacja wykonawców, stąd ostatnia jego warstwa w pełni połączona rozszczepiona była na dwie części, gdzie 27 neuronów aktywowanych funkcją *softmax* odpowiadało liczbie wykonawców do identyfikacji, zaś 84 neurony aktywowane funkcją sigmoidalną odpowiadały liczbie cech do ekstrakcji; v3.1) model identyczny do modelu v3.0, ale zadania które wykonywał wyważone były – z punktu widzenia funkcji kosztu – na korzyść identyfikacji wykonawców (koszt błędnej identyfikacji wykonawcy był dwukrotnie wyższy od kosztu błędnej ekstrakcji cechy); v4.0) model identyczny w architekturze do modelu v1.0, ale jedynym jego zadaniem była ekstrakcja cech, więc ostatnia jego warstwa w pełni połączona liczyła 84 neurony aktywowane funkcją sigmoidalną.

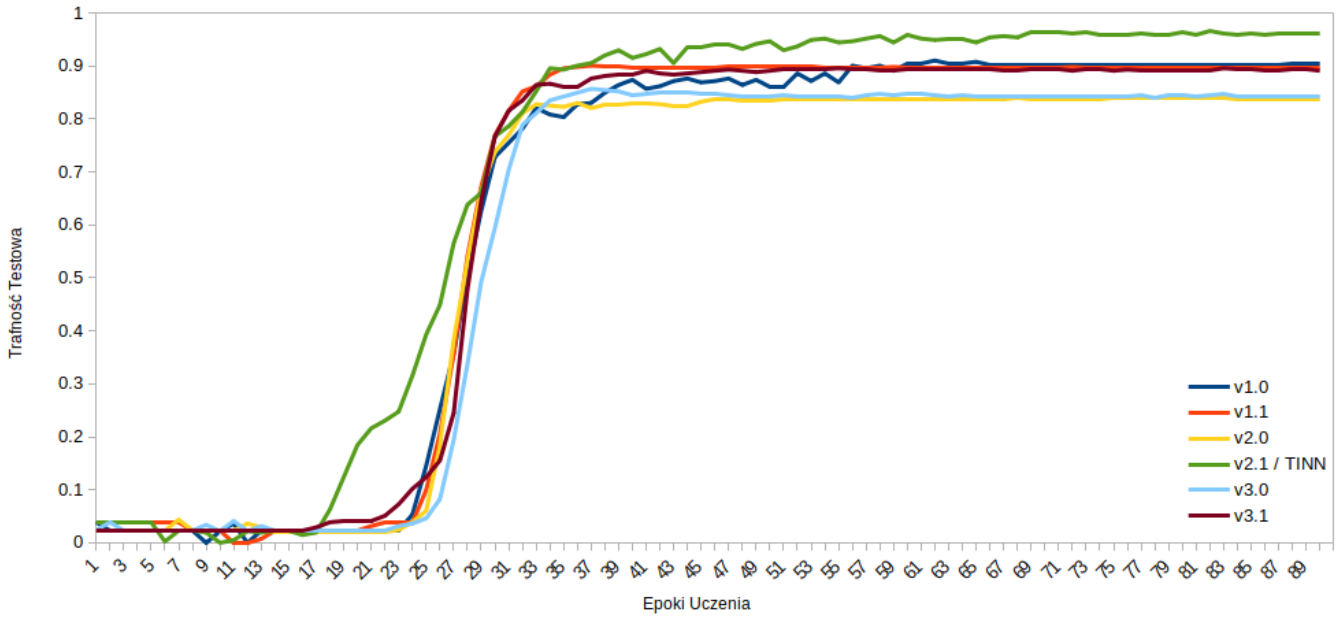
Podobnie jak w przypadku modelu TINN, co do parametrów zastosowano: i) optymizator Adam (*Adaptive Moment Estimation*) z rekomendowanymi parametrami[282]; ii) automatyczne redukcje współczynnika uczenia poprzez współczynnik redukcji wynoszący 0.1 (wykonywano iloczyn współczynnika uczenia i redukcji, gdy koszt testowy nie poprawił się przez pięć epok z rzędu); iii) *batch-size* wynosił 32; iv) prawdopodobieństwo zablokowania sygnału na warstwie opuszczającej wahało się od 0.0 do 0.25; v) trening trwał 90 epok (rys. 7.2.5, 7.2.6, 7.2.7, 7.2.8).

282 Ibid.



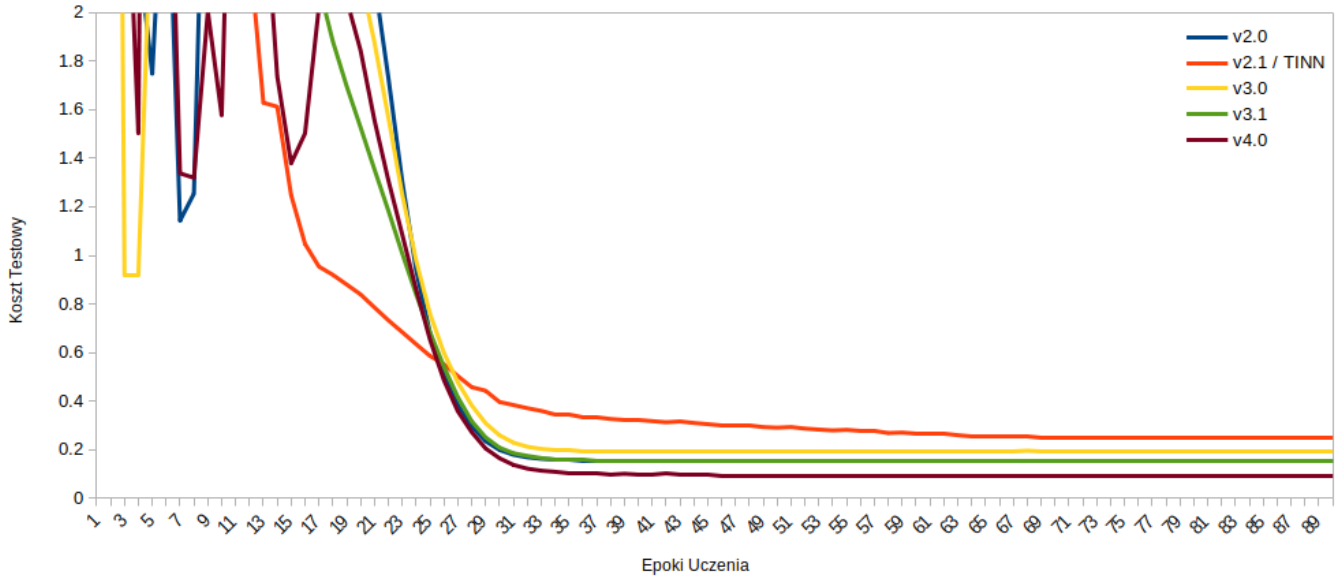
Rysunek 7.2.5. Wykres funkcji kosztu podczas uczenia modeli porównawczych – identyfikacja wykonawców.

Źródło: opracowanie własne.

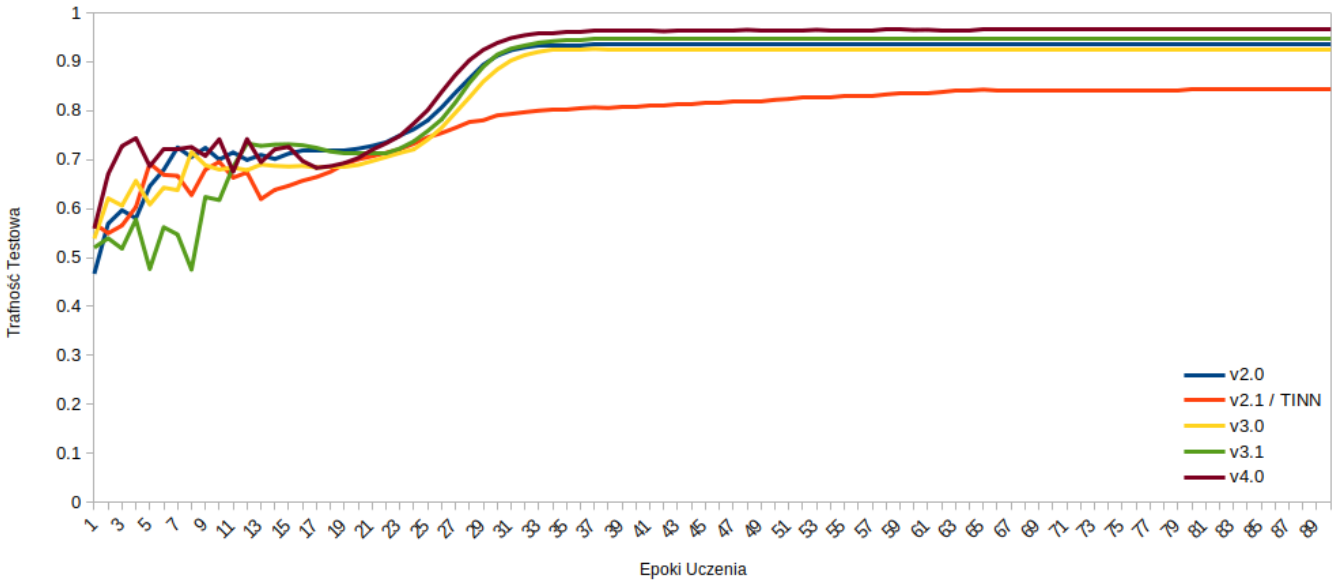


Rysunek 7.2.6. Wykres trafności podczas uczenia modeli porównawczych – identyfikacja wykonawców.

Źródło: opracowanie własne.



Rysunek 7.2.7. Wykres funkcji kosztu podczas uczenia modeli porównawczych – ekstrakcja cech.
Źródło: opracowanie własne.

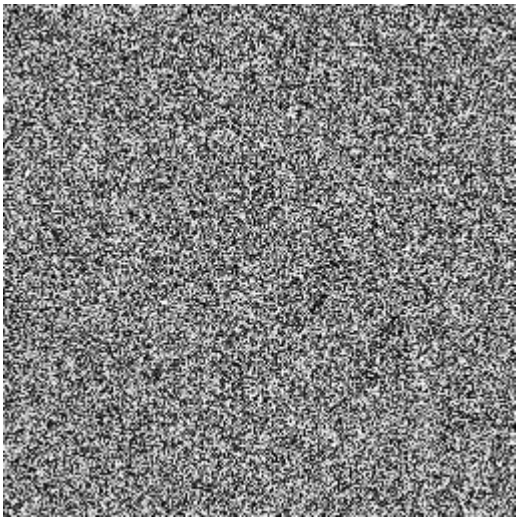


Rysunek 7.2.8. Wykres trafności podczas uczenia modeli porównawczych – ekstrakcja cech.
Źródło: opracowanie własne.

Wizualizacje. Aby ustalić jakie cechy są rzeczywiście ekstraktowane przez model, dokonano wizualizacji cech względem wyjściowych warstw modelu (*i.e.* wizualizowano cechy skorelowane z klasami wykonawców i pozytywnymi klasami cech). Należy przy tym zaznaczyć, że wizualizacje również mogą być

nieinterpretowalne, a nawet kiedy są, to nie oznacza to, że nazwy które są im nadawane pokrywają się z rozumowaniami maszyny[283].

Ogólnie rzecz biorąc, aby zwizualizować co pobudza dowolny neuron sieci[284], należy: i) przepropagować przez model losowo wygenerowany obraz (rys. 7.2.9); ii) obliczyć jakie zmiany wartości pikseli powinny zwiększyć wartość wyjściową z neuronu (*i.e.* propagować wstecznie / obliczyć gradient); 3) nanieść te zmiany/gradienty na piksele obrazu. Jest to proces analogiczny do tego, który stosuje się podczas nauki modelu, gdzie: i) propaguje się dany obraz poprzez model; ii) oblicza się jakie modyfikacje parametrów sieci powinny zbliżyć pobudzenia neuronów do oczekiwanych wartości[285] (*i.e.* oblicza się gradient funkcji $\mathrm{sieć\text{-}neuronowa}$ względem argumentu $\mathrm{dane\text{-}wejsciowe}$); iii) nanosi się te modyfikacje/gradienty na połączenia synaptyczne. W przypadku wizualizacji nie dokonuje się aktualizacji parametrów sieci, ale aktualizacji losowego obrazu wejściowego w takim kierunku, aby powodował jak najwyższe pobudzenie danego neuronu.



Rysunek 7.2.9. Przykład obrazu gdzie wartości pikseli są losowane z przedziału [0, 255].

Źródło: opracowanie własne.

283 C. Rudin, *Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead*, „Nature Machine Intelligence" t. 1 nr 5 (2019), DOI: 10.1038/s42256-019-0048-x.

284 A. Mordvintsev, C. Olah, L. Schubert, *Feature Visualization; How neural networks build up their understanding of images*, „Distill" (2017), DOI: 10.23915/distill.00007.

285 Na przykład, jeżeli propaguje się przez model dokument napisany przez pierwszego probanta, a pierwszy neuron odpowiada pierwszemu probantowi, to neuron ten powinien osiągnąć najwyższą wartości, podczas gdy pozostałe neurony powinny osiągnąć wartość minimalną.

Ogółem, zastosowano dwie metody wizualizacji, jedną podstawową[286] i jedną bardziej zaawansowaną[287]. Jedynie metoda podstawowa dała satysfakcjonujące rezultaty względem wizualizacji cech odpowiadających identyfikowanym wykonawcom. Podczas gdy, tylko metoda zaawansowana dała satysfakcjonujące rezultaty względem wizualizacji cech odpowiadających ekstraktowanym cechom pisma. Metoda podstawowa polegała na czym następuje: i) generowany jest obraz losowy; ii) losowy lub zaktualizowany obraz propagowany jest poprzez model; iii) gradienty danego neuronu obliczane są względem obrazu propagowanego poprzez model; iv) gradienty są normalizowane (a w zasadzie standaryzowane – *i.e.* różnica gradientów i ich średniej dzielona jest przez ich odchylenie standardowe – stąd ich średnia bliska jest zeru, a odchylenie standardowe jeden); v) obliczany jest iloczyn gradientów i współczynnika uczenia; vi) gradienty dodawane są do obrazu (obraz jest aktualizowany); vii) powtórzone zostają punkty ii-vii (jedna iteracja). Natomiast, metoda bardziej zaawansowana uwzględniała dodatkowe kroki, otóż: i) rozmycie gaussowskie (*gaussian blurring*) gradientów po ich normalizacji (gdzie współczynnik σ – odchylenie standardowe – redukowany był w równych krokach, raz na każdą iterację, więc z każdą iteracją rozmycie było mniejsze o tą samą wartość); ii) aplikowania gradientów dokonywał optymizator Adam; iii) po aktualizacji obrazów stosowano wobec nich normalizację L_2. Metoda podstawowa wymagała 1000 iteracji dla każdego neuronu warstwy ostatniej, przy współczynniku uczenia wynoszącym 0.1. Metoda zaawansowana wymagała 5000 iteracji dla każdego neuronu warstwy przedostatniej, przy współczynniku uczenia wynoszącym 0.00001, gdzie początkowa σ wynosiła 2.5, zaś końcowa 0.01 (wielkość filtru gaussowskiego wynosiła trzykrotność danej σ, a przynajmniej 2). Przetestowano również oktawowe skalowanie obrazów – gdzie, po danej liczbie iteracji obrazy skalowane są do większych rozmiarów, a następnie przycinane do rozmiaru oryginalnego – ale rezultaty nie były satysfakcjonujące.

286 A. Mordvintsev, C. Olah, L. Schubert, *Feature Visualization; How neural networks build up their understanding of images*, „Distill" (2017), DOI: 10.23915/distill.00007.

287 A. Øygard, *Visualizing GoogLeNet Classes* [na:] https://www.auduno.com/2015/07/29/visualizing-googlenet-classes/, 2015 r., dostęp 14 marca 2022 r.

Terminologia. Posługiwano się następującymi miarami rezultatów: i) *Loss* (lub *Loss Function*), koszt dany funkcją kosztu; ii) Acc (*Accuracy*), trafność lub dokładność; iii) AUC (*Area Under the ROC Curve*), pole powierzchni pod krzywą ROC (*Receiver Operating Characteristic*)[288].

7.3. Rezultaty i dyskusja. Ogółem, model TINN uzyskał najlepsze rezultaty w zakresie identyfikacji wykonawców i najgorsze rezultaty w zakresie ekstrakcji cech (tab. 7.3.1). Pod względem identyfikacji wykonawców, drugim modelem był v1.0 (było to jedyne zadanie wykonywane przez ten model). Pod względem ekstrakcji cech, pierwszym modelem był v4.0 (było to jedyne zadanie wykonywane przez ten model). Interesujący jest też fakt, że modele porównawcze, które wykonywały obydwa zadania – ale w ich przypadku zadanie identyfikacji nie było bezpośrednio uzależnione od zadania ekstrakcji (v2.0, v3.0, v3.1) – osiągały znacznie lepsze rezultaty w przypadku ekstrakcji niż identyfikacji. Było tak również wtedy, gdy: i) zadania wyważone były na korzyść identyfikacji wykonawców (v3.1); ii) informacja z warstwy ekstrakcyjnej wpływać mogła na decyzje podejmowane przez warstwę identyfikującą, ale nie była ich bezpośrednim determinantem (v2.0).

Najbardziej prawdopodobnym wyjaśnieniem dla: i) wyższych wyników ekstrakcji niż identyfikacji; ii) wyższych wyników ekstrakcji przez modele porównawcze niż model TINN. Jest to, że zadanie ekstrakcji nie było obostrzone zadaniem identyfikacji (nie było jego wyłącznym determinantem). Dlatego też, modele porównawcze nie były karane za przewidywanie dowolnych cech obrazów jako cech pisma. W przypadku modelu TINN, cechy ekstraktowane przez model musiały być mu użyteczne w zadaniu identyfikacji, nie mogły być więc dowolne, oraz musiały zbiegać się z oczekiwanymi cechami pisma. Warto odnotować tutaj, iż oczekiwanie, że model dokona ekstrakcji cech pisma na podstawie wytycznych – ucieleśnionych jako wektory zer i jedynek – jest próżne (nie licząc sytuacji, gdzie dysponuje się ogromnymi zasobami danych, *i.e.* obrazów pisma i opisów ich cech). Modele zawsze dążyć będą do najszybszego i najtańszego rozwiązania, ponieważ taka jest istota metody gradientowej. Sukces modelu TINN polega na bezpośrednim związku przyczynowo-skutkowym, jaki

288 Krzywa ROC obrazuje stosunek czułości do błędu pierwszego rodzaju dla różnych progów decyzyjnych klasyfikatora. Natomiast AUC określa jak dobrze model zdolny jest klasyfikować, niezależnie od wartości progów decyzyjnych.

łączy ekstrakcję i identyfikację, które wzajemnie się obostrzają. Oczywiście, może być i tak, że model ten nie wyuczył się ani ekstraktować cech pisma ani identyfikować wykonawców, ponieważ odnalazł jakieś nieznane cechy, które skorelowane są z rozwiązaniami zadań. Niemniej, bardziej jest to prawdopodobne wobec modeli porównawczych, niż modelu TINN.

Model	Epoka Uczenia	Zadanie	Koszt Treningowy	Trafność Treningowa	AUC Treningowe	Koszt Testowy	Trafność Testowa	AUC Testowe
v1.0	62	Identyfikacja Wykonawcy	0.0170	1.0000	1.0000	0.2989	0.9102	0.9904
v1.1	37	Identyfikacja Wykonawcy	0.2255	0.9665	0.9998	0.3695	0.9005	0.9977
v2.0	68	Identyfikacja Wykonawcy	0.1266	0.9738	0.9999	0.4265	0.8398	0.9965
		Ekstrakcja Cech	0.1426	0.9431	0.9888	0.1534	0.9353	0.9852
TINN / v2.1	82	**Identyfikacja Wykonawcy**	0.1166	1.0000	1.0000	**0.2422**	**0.9660**	**0.9998**
		Ekstrakcja Cech	0.2401	0.8322	0.9601	0.2477	0.8430	0.9602
v3.0	37	Identyfikacja Wykonawcy	0.2562	0.9203	0.9984	0.3891	0.8568	0.9970
		Ekstrakcja Cech	0.2173	0.9133	0.9729	0.1936	0.9262	0.9806
v3.1	55	Identyfikacja Wykonawcy	0.0503	0.9948	1.0000	0.3430	0.8956	0.9969
		Ekstrakcja Cech	0.1571	0.9448	0.9886	0.1503	0.9477	0.9888
v4.0	72	**Ekstrakcja Cech**	0.0456	0.9894	0.9994	**0.0912**	**0.9654**	**0.9938**

Tabela 7.3.1. Rezultaty ewaluacji modelu TINN i modeli porównawczych.

Źródło: opracowanie własne.

W celu upewnienia się, że model TINN nauczył się ekstraktować cechy pisma i identyfikować wykonawców, przeprowadzone zostały wizualizacje cech pobudzających neurony odpowiadające cechom pisma i tożsamości wykonawców.

Mając na względzie wizualizacje cech związanych z tożsamością wykonawców (tab. 7.3.2 i 7.3.3), zinterpretować je można jako cechy racjonalne i skorelowane z cechami pisma. W oparciu o te wizualizacje, nie byłoby uprawnionym stwierdzenie, że są one tożsame z cechami pisma (a jedynie, że są z nimi skorelowane). Mając na uwadze mechanizm działania filtrów konwolucyjnych, te zwizualizowane cechy maszynowe skorelowane być mogą *i.a.*: i) z polami i obrysami wyrazów; ii) układem wierszy względem siebie; iii) kształtem linii wierszy; iv) kierunkiem linii wierszy; v) nachyleniem pisma; vi) zmiennością nachylenia pisma; vii) wielkością pisma; viii) odstępami między wierszami; ix) odstępami między wyrazami; x) proporcjami

wysokości elementów nadlinijnych do wysokości elementów śródlinijnych; xi) proporcjami wysokości elementów podlinijnych do wysokości elementów śródlinijnych; xii) siłą nacisku i cieniowania. Interpretacja taka jest dalej uzasadniona podobieństwem wizualizacji i sum fragmentów obrazów pisma tych wykonawców. Przy czym, uzasadnienie to polega na ludzkich granicach poznawczych, takich, że na podstawie sum obrazów pisma wywnioskować by można niewiele ponad cechy wyliczone powyżej, podczas gdy sumy te są wysoce podobne do wizualizacji cech skorelowanych z tożsamościami wykonawców. Innymi słowy, nawet jeżeli nie wiemy co wynika z wizualizacji, ale są one podobne do sum, a z sum wywnioskować możemy tylko pewne cechy, to być może cechy te powinny też wynikać z wizualizacji, a przynajmniej jest to granica tego, co w wizualizacjach można nazwać.

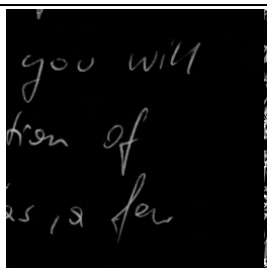
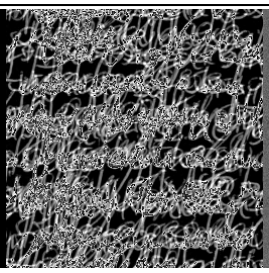
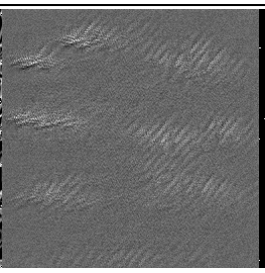
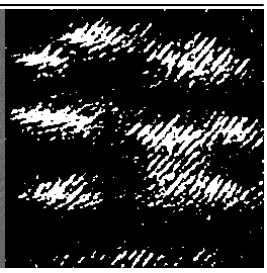
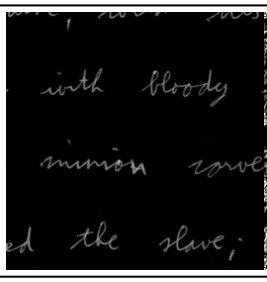
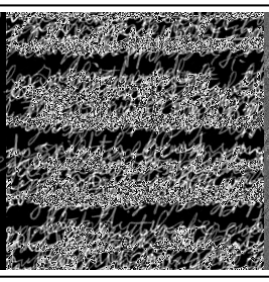
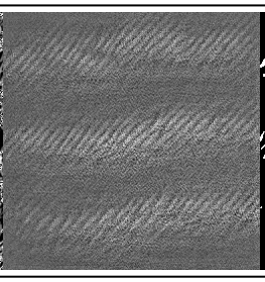
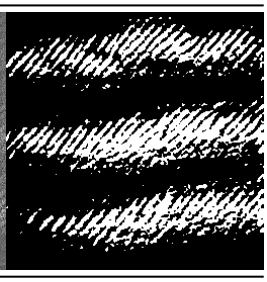
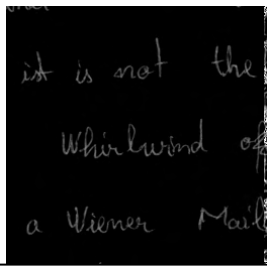
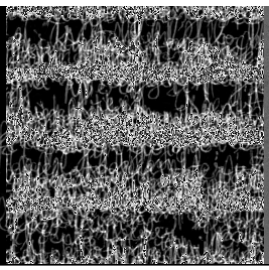
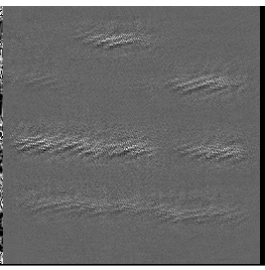
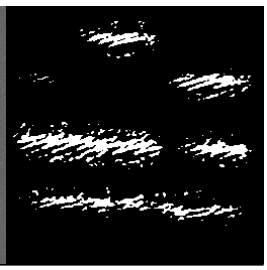
Identyfikator Wykonawcy	Przykładowy Fragment (preprocesowany)	Suma Fragmentów (preprocesowanych)	Wizualizacja Metodą Podstawową (w skali szarości)	Wizualizacja Metodą Podstawową (binaryzowana)
0005				
0006				
0015				

Tabela 7.3.2. Wizualizacje najbardziej reprezentatywnych cech związanych z tożsamościami wykonawców, uwzględniając różne metody wizualizacji (model TINN).

Źródło: opracowanie własne.

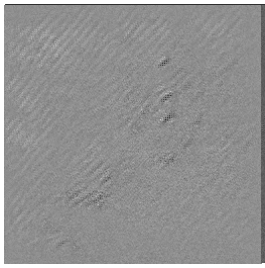
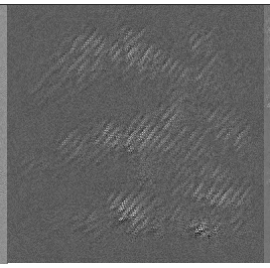
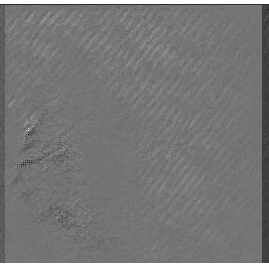
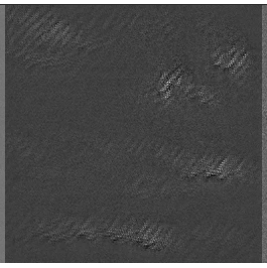
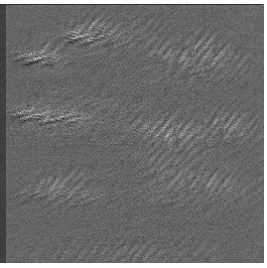
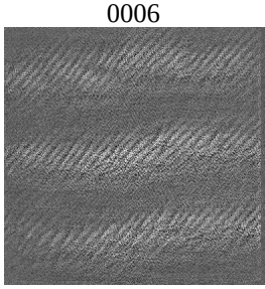
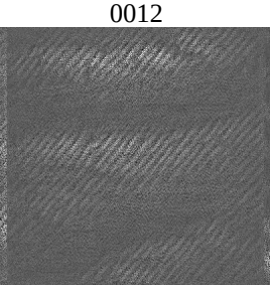
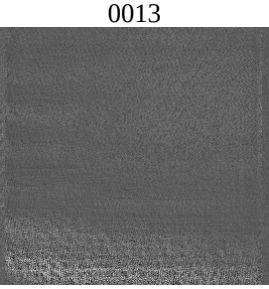
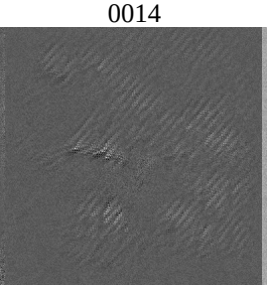
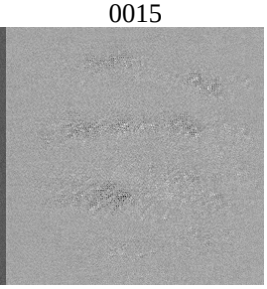
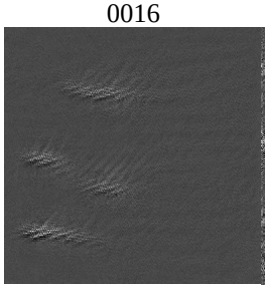
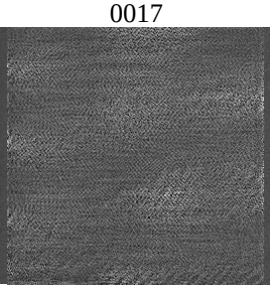
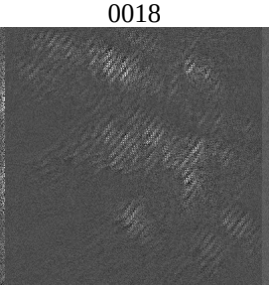
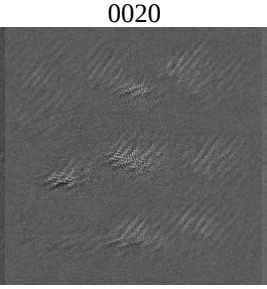
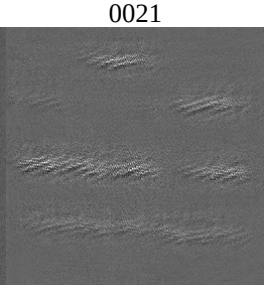
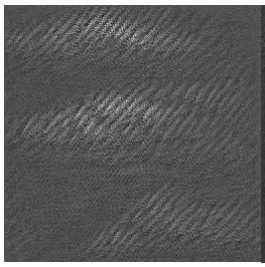
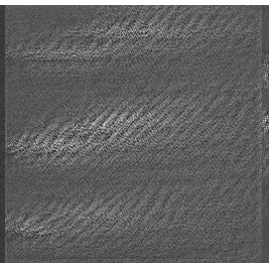
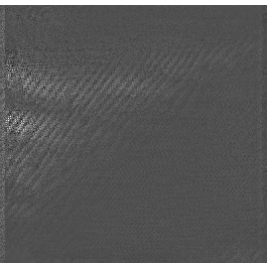
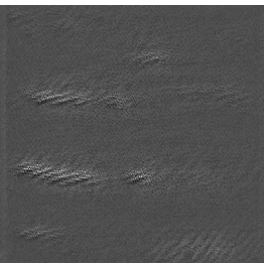
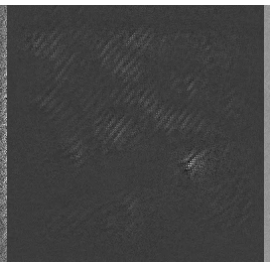
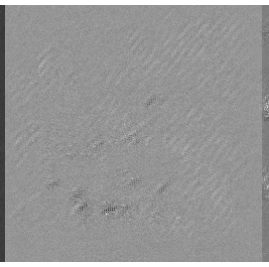
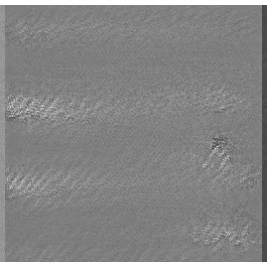
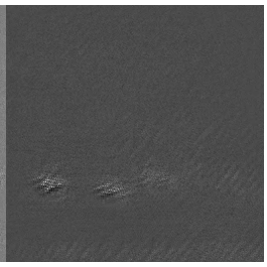
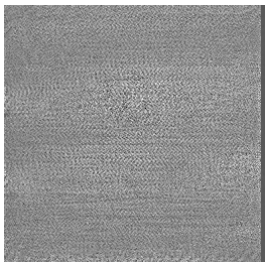
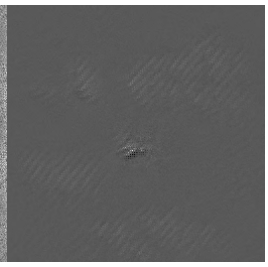
0001	0002	0003	0004	0005
				
0006	0012	0013	0014	0015
				
0016	0017	0018	0020	0021
				
0022	0023	0024	0025	0026
				
0027	0028	0029	0041	0042
				
0047	0050			
				

Tabela 7.3.3. Wizualizacje cech związanych z tożsamościami wykonawców (model TINN). Wykonane metodą podstawową i w skali szarości. Ponad wizualizacjami znajdują się identyfikatory wykonawców, których wizualizacje dotyczą.

Źródło: opracowanie własne.

Podczas gdy cechy związane z tożsamościami wykonawców sugerują korelacje z cechami pisma. To wizualizacje ekstraktowanych przez model cech związanych z cechami pisma (tab. 7.3.4 i 7.3.5) trudniej zinterpretować jako racjonalne i rzeczywiście skorelowane z cechami pisma.

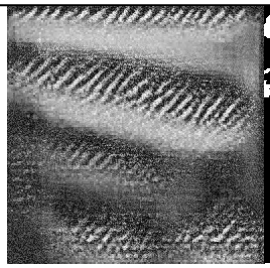
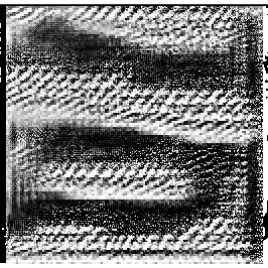
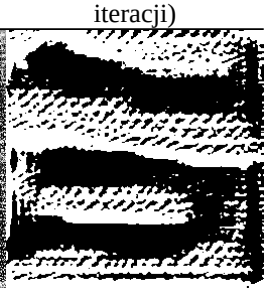
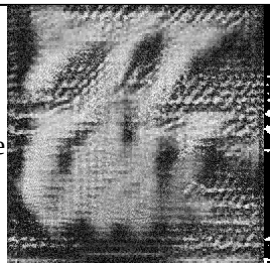
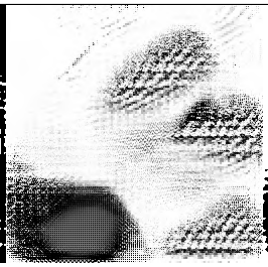
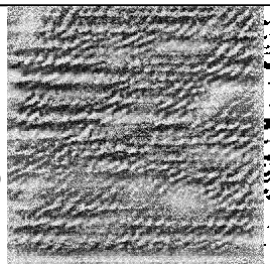
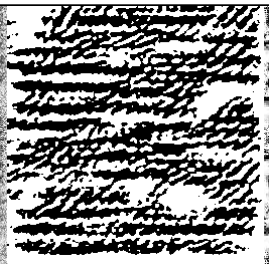
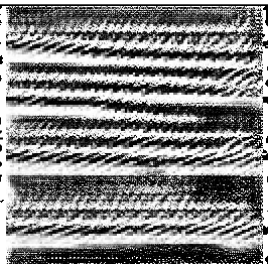
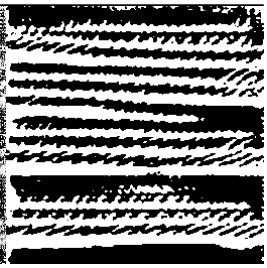
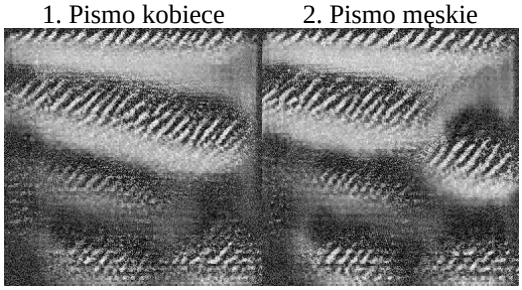
Cecha Pisma	Wizualizacja Metodą Zaawansowaną (w skali szarości; 500 iteracji)	Wizualizacja Metodą Zaawansowaną (binaryzowana; 500 iteracji)	Wizualizacja Metodą Zaawansowaną (w skali szarości; 5000 iteracji)	Wizualizacja Metodą Zaawansowaną (binaryzowana; 5000 iteracji)
Kategoria 01 Pismo Kobiece				
Cecha 35 Jednorodne Nachylenie Pisma				
Cecha 56 Elementy Poziome Kreślone od Prawej do Lewej				

Tabela 7.3.4. Wizualizacje przykładowych cech związanych z cechami pisma, uwzględniając różne metody wizualizacji (model TINN).

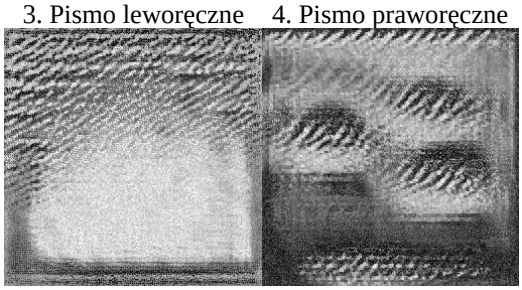
Źródło: opracowanie własne.

Kategorie (zbiory cech pisma)

1. Pismo kobiece 2. Pismo męskie



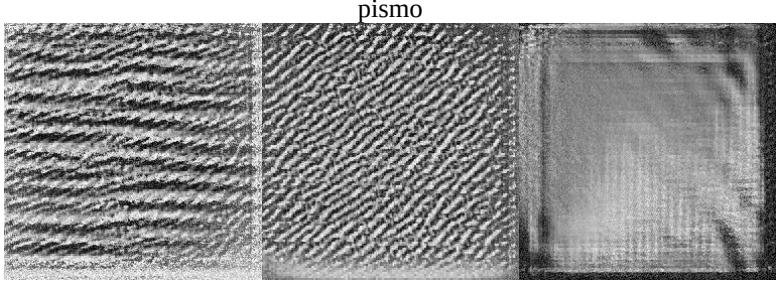
3. Pismo leworęczne 4. Pismo praworęczne



Cechy Pisma

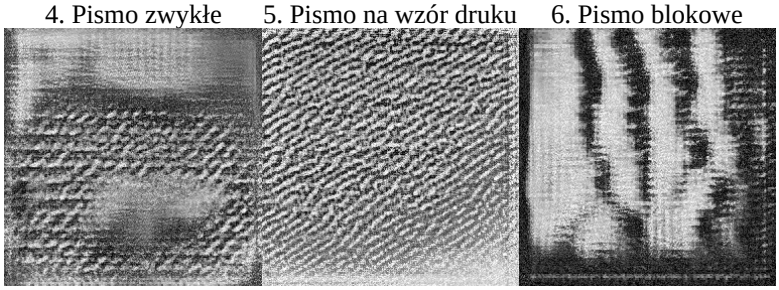
I. Wyrobienie pisma

1. Wyrobione pismo 2. Średnio wyrobione pismo 3. Niewyrobione pismo



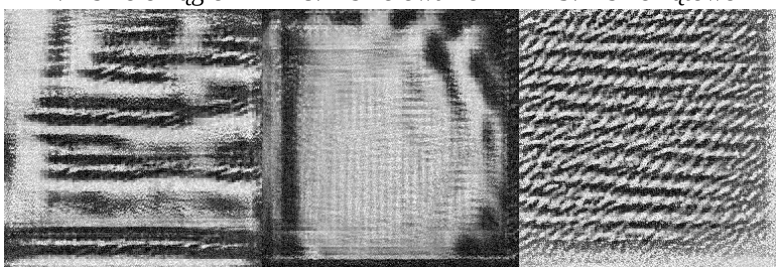
II. Typ pisma

4. Pismo zwykłe 5. Pismo na wzór druku 6. Pismo blokowe



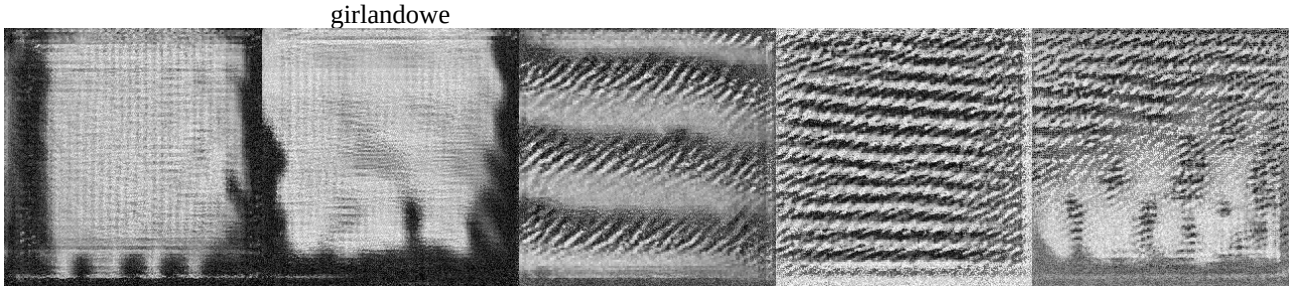
III. Ogólny obraz pisma

7. Pismo okrągłe 8. Pismo owalne 9. Pismo kątowe



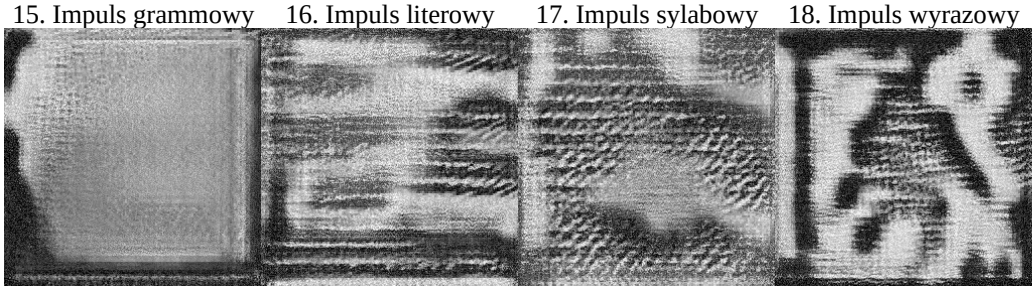
IV. Budowa wiązań międzyliterowych

10. Wiązania arkadowe | 11. Wiązania girlandowe | 12. Wiązani kątowe | 13. Wiązania stykowe | 14. Brak wiązań



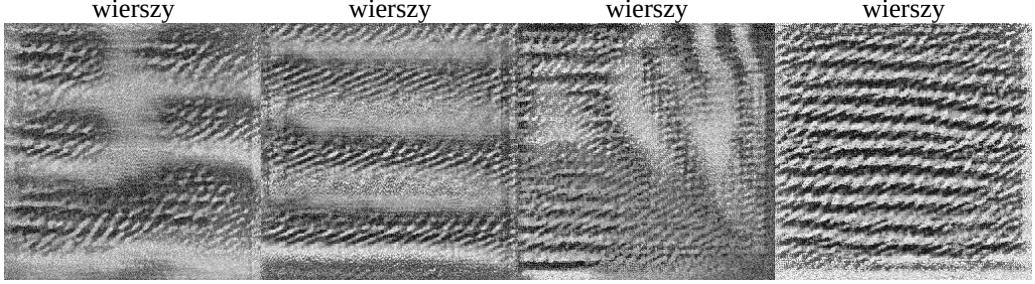
V. Impuls dominujący

15. Impuls grammowy | 16. Impuls literowy | 17. Impuls sylabowy | 18. Impuls wyrazowy



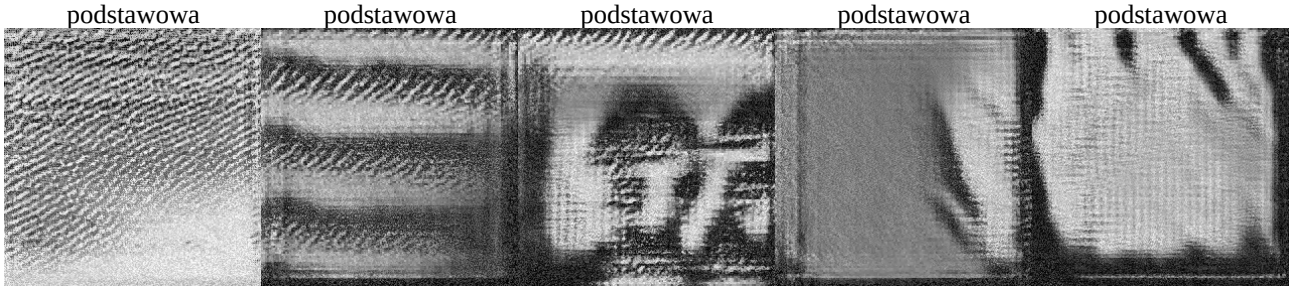
VI. Układ wierszy względem siebie

19. Równoległy układ wierszy | 20. Rozbieżny układ wierszy | 21. Zbieżny układ wierszy | 22. Nieregularny układ wierszy



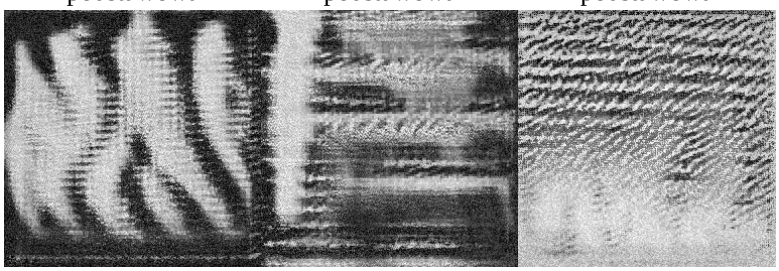
VII. Kształt linii wierszy

23. Prosta linia podstawowa | 24. Łamana linia podstawowa | 25. Falista linia podstawowa | 26. Arkadowa linia podstawowa | 27. Girlandowa linia podstawowa



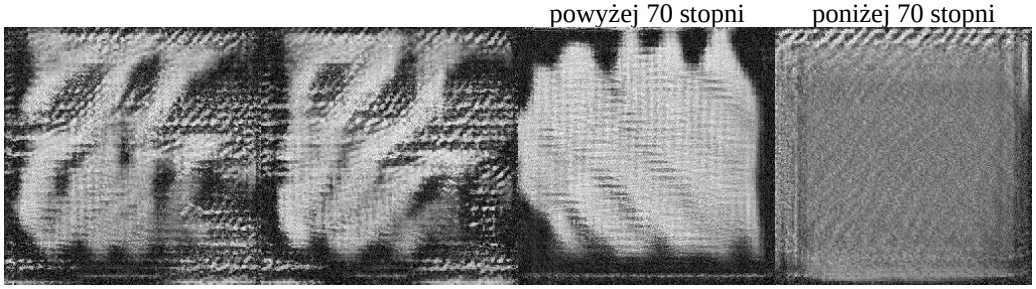
VIII. Kierunek linii wierszy

28. Wznosząca się linia podstawowa | 29. Pozioma linia podstawowa | 30. Opadająca linia podstawowa



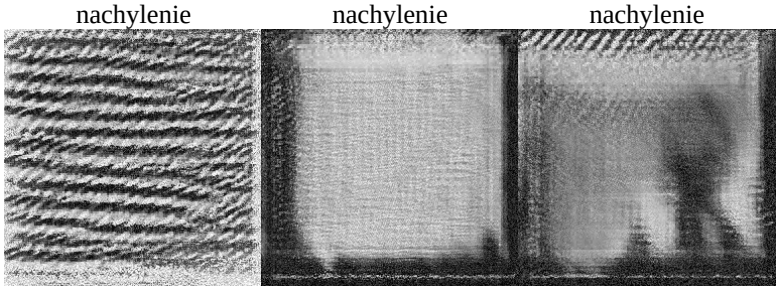
IX. Nachylenie pisma

31. Pismo lewoskośne | 32. Pismo proste | 33. Pismo prawoskośne powyżej 70 stopni | 34. Pismo prawoskośne poniżej 70 stopni



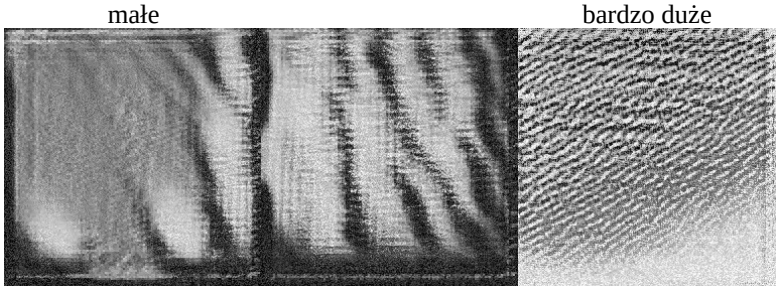
X. Zmienność nachylenia pisma (*quality of the slant*)

35. Jednorodne nachylenie | 36. Zmienne nachylenie | 37. Naprzemienne nachylenie



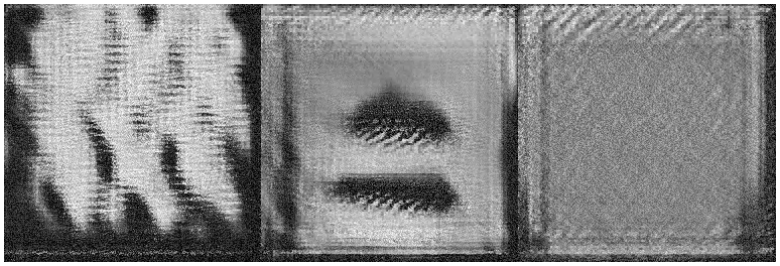
XI. Wielkość pisma

38. Pismo drobne lub małe | 39. Pismo średnie | 40. Pismo duże i bardzo duże



XII. Odstępy między wierszami

41. Zmniejszone odstępy między wierszami | 42. Wzorcowe odstępy między wierszami | 43. Powiększone odstępy między wierszami

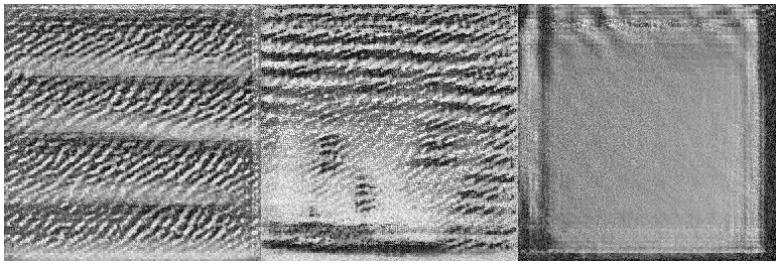


XIII. Odstępy między wyrazami

44. Zmniejszone odstępy między wyrazami

45. Przeciętne odstępy między wyrazami

46. Powiększone odstępy między wyrazami

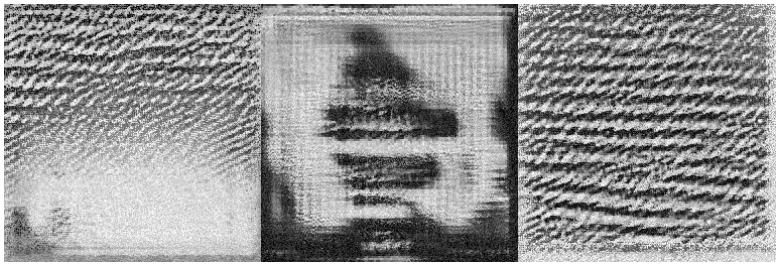


XIV. Odstępy między znakami

47. Zmniejszone odstępy między znakami

48. Wzorcowe odstępy między znakami

49. Powiększone odstępy między znakami

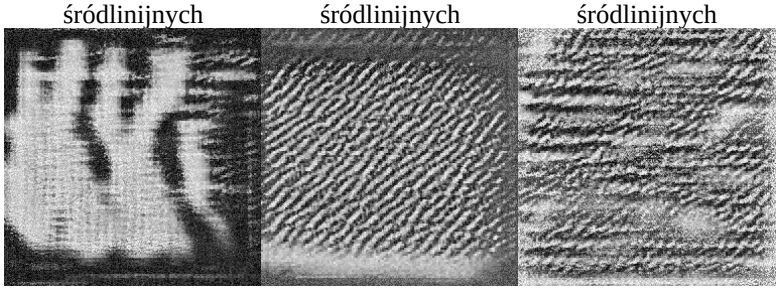


XV. Proporcje wysokości elementów nadlinijnych do wysokości elementów śródlinijnych

50. Elementy nadlinijne zmniejszone wobec śródlinijnych

51. Elementy nadlinijne proporcjonalne do śródlinijnych

52. Elementy nadlinijne powiększone wobec śródlinijnych

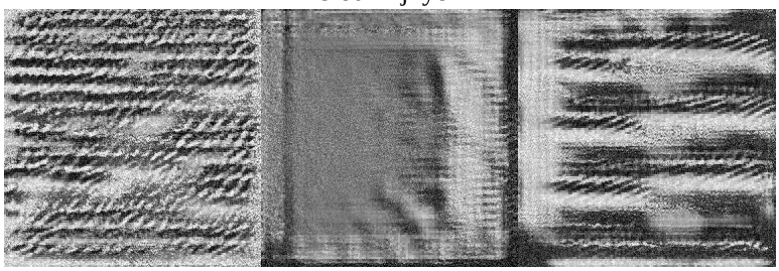


XVI. Proporcje wysokości elementów podlinijnych do wysokości elementów śródlinijnych

53. Elementy podlinijne zmniejszone wobec śródlinijnych

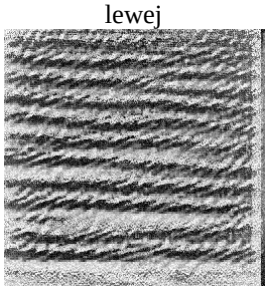
54. Elementy podlinijne proporcjonalne do śródlinijnych

55. Elementy podlinijne powiększone wobec śródlinijnych

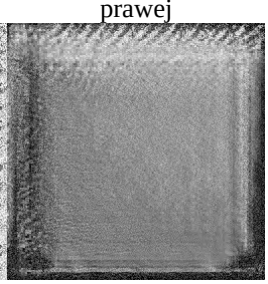


XVII. Kierunek kreślenia elementów poziomych

56. Elementy poziome kreślone od prawej do lewej

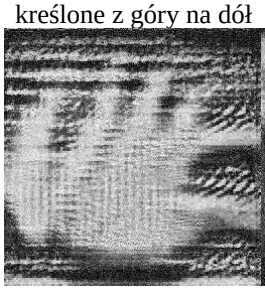


57. Elementy poziome kreślone od lewej do prawej



XVIII. Kierunek kreślenia elementów pionowych

58. Elementy pionowe kreślone z góry na dół

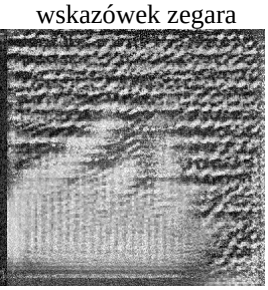


59. Elementy pionowe kreślone z dołu na góry

XIX. Kierunek kreślenia owali, łuków i pętlic

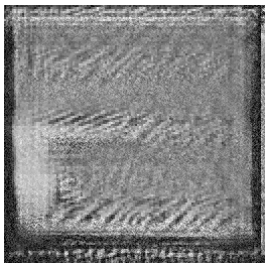
60. Owale kreślone zgodnie z ruchem wskazówek zegara

61. Owale kreślone przeciwnie do ruchu wskazówek zegara

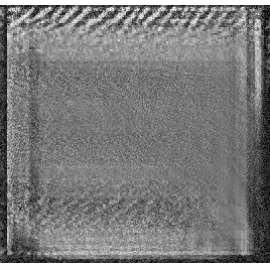


XX. Kierunek kreślenia inicjacji

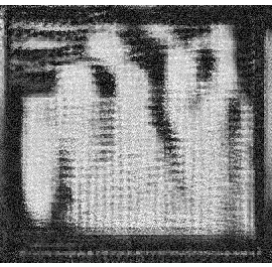
62. Inicjacje kreślone do góry i w lewo



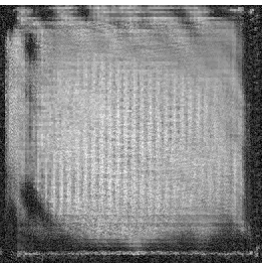
63. Inicjacje kreślone do góry i w prawo



64. Inicjacje kreślone do dołu i w lewo

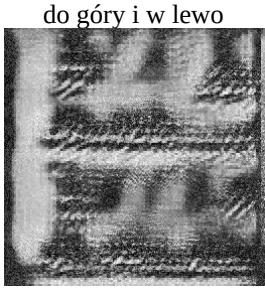


65. Inicjacje kreślone do dołu i w prawo



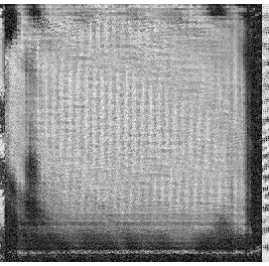
XXI. Kierunek kreślenia terminacji

66. Terminacje kreślone do góry i w lewo

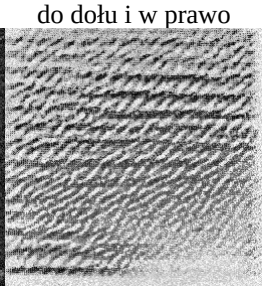


67. Terminacje kreślone do góry i w prawo

68. Terminacje kreślone do dołu i w lewo

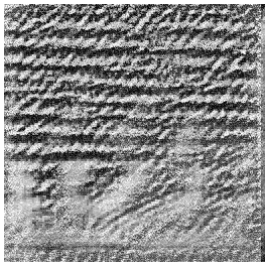


69. Terminacje kreślone do dołu i w prawo

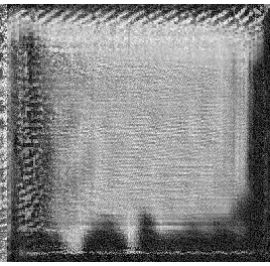


XXII. Siła nacisku i cieniowania

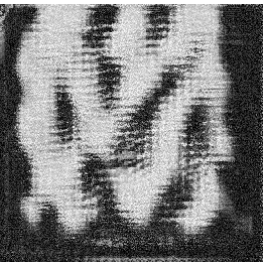
70. Duża siła nacisku



71. Średnia siła nacisku



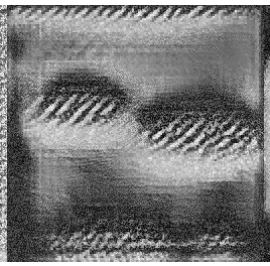
72. Mała siła nacisku



XXIII. Kierunek nacisku i cieniowania

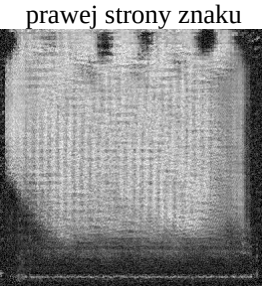
73. Nacisk wzmożony na górze

74. Nacisk wzmożony na dole



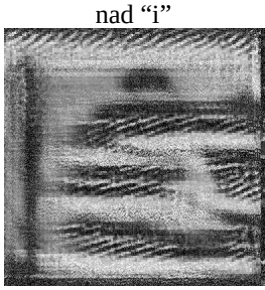
75. Nacisk wzmożony z lewej strony znaku

76. Nacisk wzmożony z prawej strony znaku



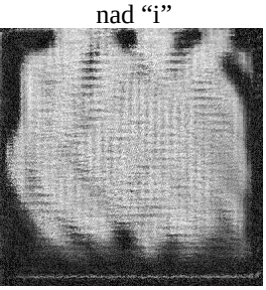
XXIV. Znaki diakrytyczne

77. Kreski diakrytyczne nad “i”



78. Kropki diakrytyczne nad “i”

79. Owale diakrytyczne nad “i”



XXV. Manieryzmy i ozdoby

80. Manieryzmy i ozdoby

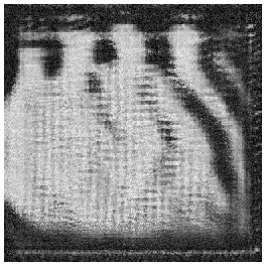
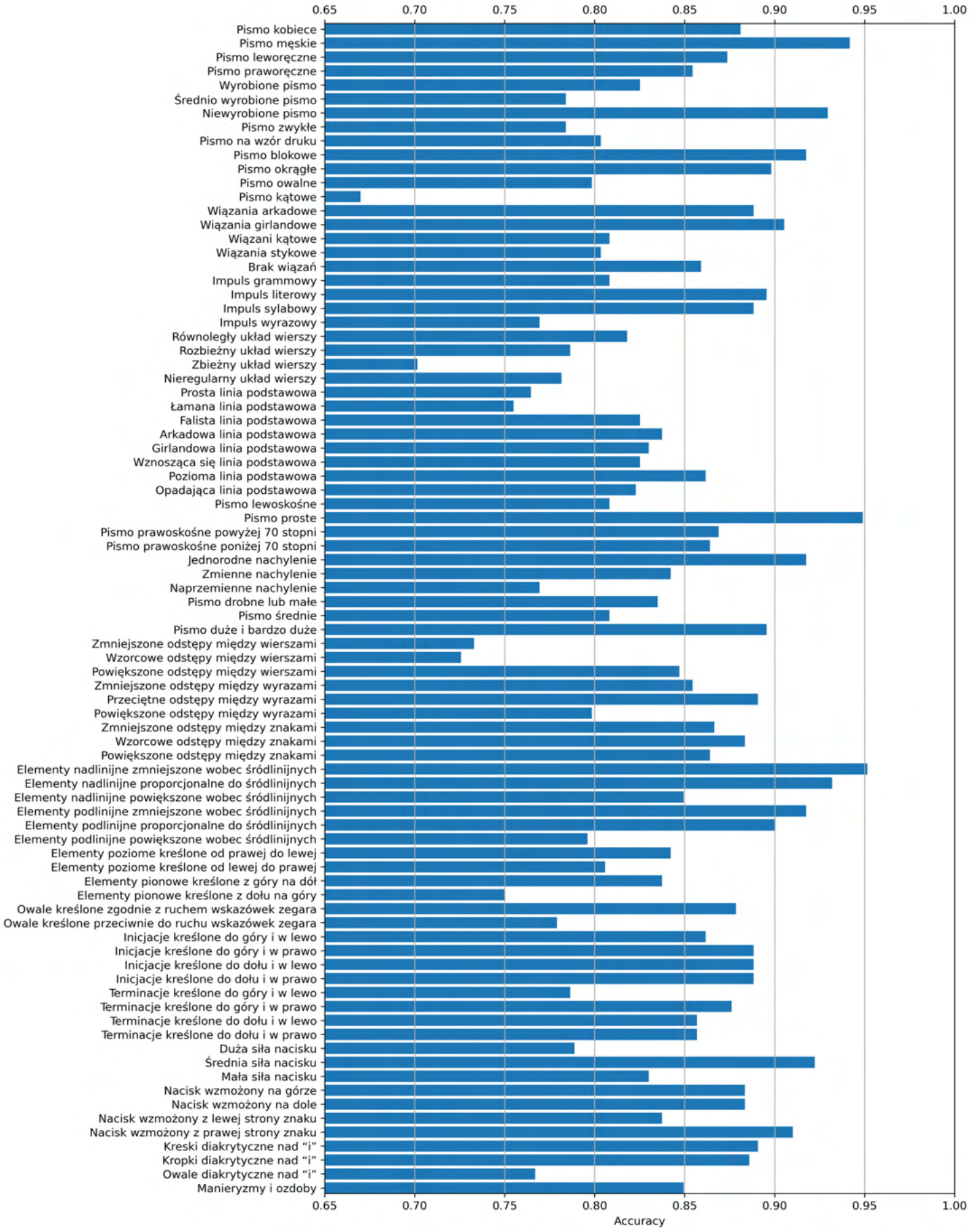


Tabela 7.3.5. Wizualizacje cech związanych z cechami pisma (model TINN). Wykonane metodą zaawansowaną w 500 iteracjach i skali szarości. Ponad wizualizacjami znajdują się nazwy kategorii i cech, których wizualizacje dotyczą.
Źródło: opracowanie własne.

Jednakże, oczekiwano od modelu, aby nauczył się większej liczby cech (rys. 7.3.1), niż uprzednio wymienione. Ponadto, jeżeli te uprzednio wyliczone cechy zostały opanowane przez model, to powinien on osiągnąć większą trafność podczas ich ekstrakcji, niż podczas ekstrakcji pozostałych cech. Podczas gdy model osiągnął następujące średnie trafności dla tych cech: i) układ wierszy względem siebie – 0.7719; ii) kształt linii wierszy – 0.8025; iii) kierunek linii wierszy – 0.8366; iv) nachylenie pisma – 0.8940; v) zmienność nachylenia pisma – 0.8430; vi) wielkość pisma – 0.8463; vii) odstępy między wierszami – 0.7686; xviii) odstępy między wyrazami – 0.8479; ix) proporcje wysokości elementów nadlinijnych do wysokości elementów śródlinijnych – 0.9110; x) proporcje wysokości elementów podlinijnych do wysokości elementów śródlinijnych – 0.8713; xi) siła nacisku i cieniowania – 0.8470. Ogólna średnia z tych rezultatów wyniosła 0.8400, jest więc nieistotnie różna od ogólnej średniej trafności ekstrakcji cech pisma przez model, która wyniosła 0.8430. Przypuszczać można, że: i) albo model nauczył się cech pisma, ale nie są zbieżne ze swoimi etykietami bądź nazwami; ii) albo model w ogóle nie nauczył się cech pisma. Autor jest tutaj zdania, że wariant pierwszy jest bardziej prawdopodobny.

Model przetestowano także poprzez podawanie mu cech pisma opisanych przez autora wprost na jego warstwę identyfikującą/wyjściową, jako danych wejściowych na tą warstwę. Model TINN osiągnął następujące rezultaty na zbiorze testowym: i) koszt – 1.8590; ii) trafność – 0.5583; AUC – 0.9530. Podczas gdy rezultaty na zbiorze treningowym były następujące: i) koszt – 1.8730; ii) trafność – 0.5503; iii) AUC – 0.9486. Na podstawie wysokości tych rezultatów oraz ich podobieństwa, wnioskować można, że model nauczył się istotnej liczby nieznanych cech, które zbieżne są pod względem wartości i częstotliwości z oczekiwanymi cechami pisma. Mogą być te cechy ze sobą tożsame, a są przynajmniej wysoce skorelowane.



Rysunek 7.3.1. Rozkład trafności modelu TINN ze względu na ekstraktowane cechy.

Źródło: opracowanie własne.

7.4. Wnioski. Po pierwsze, zaproponowano nowe podejście do problemu interpretowalności maszyn uczących się, w postaci powierzchownie interpretowalnej sieci neuronowej (*top interpretable neural network*, TINN), gdzie: i) zadaniem modelu była identyfikacja wykonawców; ii) na podstawie cech ekstraktowanych przez model z danych surowych; iii) który uczony był pod nadzorem ekstraktować cechy pisma i wykorzystywać je jako wyłączne determinanty tożsamości wykonawców.

Po drugie, mając na względzie wielkość populacji ludzkiej, zadanie identyfikacji osób na podstawie pisma jest niezwykle trudnym problemem politomicznym (wieloklasowość, gdzie każdy człowiek stanowi klasę indywidualną), którego nie da się rozwiązać trenując sieci neuronowe do identyfikacji osób. Ponieważ, jeżeli wykonawca dokumentu kwestionowanego nie był rozważany na etapie trenowania modelu (wykonawca zewnętrzny), to jakaś inna osoba, która była rozważana na etapie trenowania modelu, może zostać przez model wskazana jako najbardziej prawdopodobny wykonawca dokumentu kwestionowanego. Najpopularniejszym sposobem na rozwiązanie problemu ogromnych politomii są modele weryfikacyjne, gdzie model porównuje dwie próbki danych i określa, czy mają wspólne źródło, czy też nie, redukując zatem problem do problemu dychotomicznego (dwuklasowego, binarnego). Przedstawiony powyżej model TINN stanowić może nowe rozwiązanie powyższego problemu. Skoro bowiem model TINN ekstraktuje cechy pisma i dopiero na tej podstawie dokonuje identyfikacji wykonawców, to jego predykcje tożsamościowe są sprawdzalne już na poziomie ekstrakcji cech, *i.e.*: czy cechy wyekstraktowane przez model z dokumentów kwestionowanych są zbieżne z cechami wyekstraktowanymi przez model z dokumentów porównawczych. Innymi słowy, gdyby zapytać nieinterpretowalny model identyfikacyjny o ustalenie wykonawcy dokumentu, który nie był rozważany na etapie trenowania modelu (wykonawca zewnętrzny), to model taki: i) dokonać może predykcji nierozstrzygającej (*i.e.* przypisać podobne prawdopodobieństwo do wszystkich wykonawców); ii) może też dokonać jednoznacznej identyfikacji (*i.e.* przypisać wysokie prawdopodobieństwo do jednego wykonawcy), która raczej nie będzie podważana, chyba że podejrzewane będzie zewnętrzne wykonawstwo dokumentu kwestionowanego. Jeżeli model TINN dokonałby jednoznacznej identyfikacji wykonawcy, a dokument był w rzeczywistości pochodzenia zewnętrznego, to podważyć można takie rozstrzygnięcie, porównując ze sobą: i) cechy

wyekstraktowane przez model z dokumentu kwestionowanego; ii) i cechy wyekstraktowane przez model z dokumentów reprezentatywnych dla wskazanego przez model wykonawcy. Gdyby cechy te nie były zbieżne, to podważyć można predykcje modelu, stwierdzając że dokonał najlepszej możliwej w danym momencie, ale fałszywej predykcji. Gdyby cechy te zaś były zbieżne, a wiadomo by było, że istnieje ryzyko, iż dokument kwestionowany jest zewnętrznego wykonawstwa, to predykcje modelu będą nieużyteczne, a należało będzie zasięgnąć opinii eksperta. Gdyby zaś cechy te były zbieżne, a nie wiedziano by, że istnieje ryzyko zewnętrznego wykonawstwa dokumentu kwestionowanego, to model wprowadziłby użytkowników w błąd. Podsumowując, model powierzchownie interpretowalny nie stanowi rozwiązania zupełnego, ale powinien być dalece rzetelniejszy w swoich predykcjach niż model nieinterpretowalny.

Po trzecie, decyzja o tym czy zawierzyć w predykcje modelu TINN nie musi polegać wyłącznie na zawierzeniu w trafność jego identyfikacji i ekstrakcji, ale może polegać przede wszystkim na analizie cech, które on ekstraktuje. Otóż, sprawdzić można, czy: i) cechy wyekstraktowane przez model zbieżne są z cechami opisanymi przez eksperta oraz czy nie są one wewnętrznie sprzeczne? ii) czy cechy wyekstraktowane przez model, które miały decydujący wpływ na rozstrzygnięcie identyfikujące, są zbieżne z cechami, które miały decydujący wpływ na rozstrzygnięcie dokonane przez eksperta? Najprostszym sposobem, aby ustalić, które spośród cech wyekstraktowanych przez model miały największy wpływ na rozstrzygnięcie identyfikujące, jest zastosowanie metody gradientowej. Otóż, obliczyć można, które spośród wyekstraktowanych cech należałoby wzmocnić, a które osłabić, aby zwiększyć prawdopodobieństwo danej predykcji. Cechy, które należałoby wzmocnić poczytać można za najsilniejsze podobieństwa, a cechy, które należałoby osłabić, poczytać można za najsilniejsze różnice. Jeżeli więc wskaźniki trafności są ogółem satysfakcjonujące, a powyższe warunki spełnione, to predykcje modelu wykorzystane mogą zostać jako argument wzmacniający opinię eksperta. W sytuacji przeciwnej, jeżeli wskaźniki trafności są ogółem satysfakcjonujące, a powyższe dwa warunki zostały spełnione tylko wobec dokumentów porównawczych, nie zaś wobec dokumentu kwestionowanego, to predykcje modelu wykorzystane mogą zostać jako argument podważający opinię eksperta. W innych zaś przypadkach model nie będzie nadawał się ani do zastosowania, ani jako argument, poza następującym wyjątkiem. Jeżeli przykładowy ekspert

przeprowadził identyfikacyjne badania pismoznawcze, to wykorzystać można jego opinię do dopracowania identyfikacyjnego modelu TINN, celem zreprodukowania rezultatów ekspertyzy i zweryfikowania czy jest ona samo-podtrzymująca-się (*self-sustainable*). Na przykład, podważać można by wyniki takiej ekspertyzy, jeżeli jej jakość byłaby niedostateczna, aby na jej podstawie nauczyć model TINN identyfikacji osób, których dotyczyła (zakładając, że ekspertyzę przeprowadzono na znacznej próbie osób podejrzanych o wykonawstwo). Zaznaczyć tutaj należy, że samo tylko skuteczne wyuczenie modelu TINN na podstawie ekspertyzy, nie może stanowić o jej trafności, a jedynie o tym, że jej rezultaty są reprodukowalne i jest ona samo-podtrzymująca-się.

Po czwarte, uwzględniając treść punktów drugiego i trzeciego, zauważyć należy, że ze względu na ich inherentną uniwersalność, weryfikacyjne modele TINN mogły by być stosowane zarówno do weryfikacji pisma, jak i do identyfikacji wykonawców, a to niezależnie od danej sprawy i eksperta. Jednakże, ponieważ istota weryfikacji jest dalece bardziej abstrakcyjna i generalna od identyfikacji, to uczenie modeli weryfikacyjnych typu TINN wymagać będzie ogromnych ilości danych, pochodzących z ogromnej liczby, a wysokiej jakości źródeł. Ponadto, podkreślić należy, że ani modele weryfikacyjne ani modele identyfikacyjne nie będą stanowić skutecznych metod badania pisma w przypadkach fałszerstwa lub maskowania nawyku, chyba że uczone będę celowo do rozwiązywania takich przypadków (w tej sytuacji koszt uczenia modeli weryfikacyjnych będzie jeszcze większy).

Po piąte, wyjaśnić należy, że cechy powinny być ekstraktowane przez model, raczej niż wprowadzane do modelu przez eksperta (jako dane wejściowe służące do identyfikacji wykonawców). Otóż, gdyby model uczono identyfikować wykonawców na podstawie cech wyekstraktowanych przez eksperta, to model taki jest zupełnie od eksperta zależny. Chociaż predykcje takiego modelu powinny być użyteczne i rzetelne, a model dowodzić reprodukowalności wyników i samo-podtrzymywania-się ekspertyzy. To, jeżeli można osiągnąć większą niezależność, należy do niej dążyć. Ponadto, model TINN może dokonywać swoich predykcji zarówno na podstawie cech, które sam ekstraktuje, jak też na podstawie cech opisanych przez eksperta.

Po szóste, rozważyć należy czy modele powierzchownie interpretowalne będą bardziej obiektywne od swoich nieinterpretowalnych odpowiedników maszynowych lub powierzchownie interpretowalnych odpowiedników ludzkich. Człowiek nabywa

obiektywności poprzez stosowanie metod naukowych, za które ręczy statystyka lub logika, nie wyklucza to jednak błędów w stosowaniu tych metod. Modele uczenia maszynowego postrzegane są często jako obiektywne, jeżeli są trafne i trenowane na danych niezależnych od ludzkiego uznania. Jednakże, skoro nie potrafimy interpretować ich procesów decyzyjnych, to nie wiemy czy ich metody zbieżne są z metodami naukowymi. Automatyzacja nie gwarantuje więc mniejszej ilości subiektywizmu ludzkiego, ale większą ilość subiektywizmu maszynowego. Podczas gdy, modele powierzchownie interpretowalne można by uznawać za obiektywne, pod warunkiem, że: i) dane do ich uczenia pochodzą od wielu niezależnych ekspertów; ii) umożliwiając im osiąganie wysokiej trafności i rzetelności; iii) podczas gdy cechy przez nie ekstraktowane i decyzje, które podejmują na ich podstawie, zbieżne są z metodami naukowymi.

Po siódme, rezultaty omawianych badań są wysoce obiecujące, pomimo że model nie nauczył się cech pisma bezpośrednio, a nauczył się dyskryminatywnych zakresów i częstotliwości występowania wysoce złożonych i zlokalizowanych cech, które wydają się być skorelowane z cechami pisma. Mając na uwadze małą skalę przeprowadzonego eksperymentu, z jednej strony jest on bliski praktyce, a z drugiej strony dowodzi, iż może to być owocny kierunek badań na większą skalę. Zdaniem autora, aby osiągnąć lepsze rezultaty, należałoby: i) pozyskać znacznie więcej materiału uczącego od większej liczby probantów; ii) wprowadzać do modelu większe fragmenty lub całe dokumenty; iii) zamiast ogólnego opisywania nawyków, które nie zawsze będą reprezentowane przez dane wejściowe, opisywać cechy pisma występujące na poszczególnych fragmentach lub dokumentach i oczekiwać od modelu ekstrakcji tylko tychże. Ponadto, może okazać się owocnym, wytrenowanie takiego modelu, a następnie podział jego na dwa sub-modele, gdzie: i) zadaniem jednego jest ekstrakcja cech występujących na poszczególnych fragmentach lub dokumentach; ii) nowym zadaniem drugiego jest identyfikacja wykonawców na podstawie ich nawyków (*i.e.* drugi sub-model wpierw identyfikował wykonawców na podstawie cech ekstraktowanych z poszczególnych fragmentów lub dokumentów, a teraz identyfikuje ich na podstawie ogółu cech wyekstraktowanych przez pierwszy sub-model ze wszystkich fragmentów lub dokumentów).

Po ósme, gdyby odrzucić model TINN, z braku przekonania, że cechy które on ekstraktuje są skorelowane z cechami pisma, to byłaby to prosta i definitywna falsyfikacja. Ponadto, falsyfikacja modelu TINN pociągałaby za sobą falsyfikację wszystkich modeli porównawczych. Nie można bowiem zaakceptować, że modele nieinterpretowalne zdolne są rozwiązywać problemy racjonalnie i rzetelnie, a tym bardziej podług metody naukowej, skoro odrzuca się podobny do nich i podobnie trafny model, który wspomagano i nadzorowano w osiąganiu racjonalności, a który nie był do niej zdolny i został sfalsyfikowany.

Podsumowując, metoda powierzchownie interpretowalnych sieci neuronowych może stać się skutecznym remedium na problemy i metodą na zastosowania sztucznych sieci neuronowych do badań kryminalistycznych. Po pierwsze, powinna umożliwiać trenowanie bardziej trafnych modeli. Po drugie, powinna umożliwiać interpretację modeli na poziomie hierarchicznie wykonywanych zadań, które są powiązane przyczynowo-skutkowo i nazwane (semantycznie sensowne). Po trzecie, istotnie podnosi poziom falsyfikowalności modeli, gdyby bowiem ocenić cechy ekstraktowane przez model jako nieprzekonywujące, to można go odrzucić niezależnie od jego poziomu trafności. Po czwarte, rezultaty powyższych badań, przeprowadzonych na małej skali problemie identyfikacyjnym, sugerują że podobna metoda mogłaby osiągnąć znacznie lepsze rezultaty na dużej skali problemie weryfikacyjnym.

Ostatecznie, jeżeli metoda modeli powierzchownie interpretowalnych nie jest przekonywująca, to tym bardziej należy odrzucić metody nieinterpretowalne.

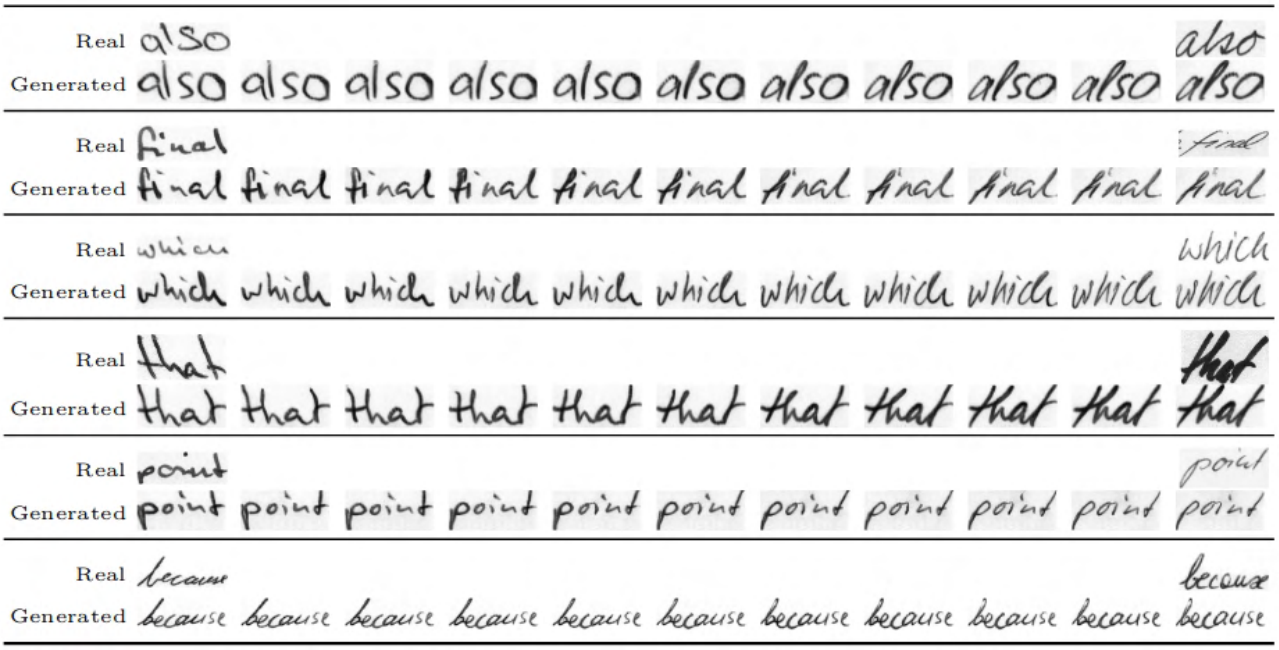
7.5. Reprodukowalność. Wszystkie szczegóły, metody, rezultaty, dane i dodatkowe objaśnienia zostały udokumentowane i dostępne są w repozytorium autora na platformie Github: https://github.com/Ma-Marcinowski/Identificational-Model

Modele uczono i testowano za pomocą chmury obliczeniowej Google Colab(oratory), umożliwiającej korzystanie z GPU Nvidia Tesla K80. Gdzie przeciętna epoka uczenia modelu trwała 120 sekund (modele trenowano po 90 epok każdy).

Rozdział 8. Przykład wykrywania fałszerstw sztucznych sieci neuronowych na przykładzie badań pismoznawczych.

8.1. Wprowadzenie. Celem zobrazowania problemu głębokich fałszerstw, autor opracował generatywno-adwersaryjną sieć neuronową do fałszowania podpisów. Podpisy tak sfałszowane autor poddał ocenie z punktu widzenia badań pismoznawczych, oraz dokonał wskazania takich cech tych podpisów, które nie są *stricte* pismoznawcze, ale które zdradzają sztuczność badanego podpisu.

Popularnym tematem zastosowań sieci generatywno-adwersaryjnych jest generowanie pisma ręcznego na podstawie tekstu komputerowego. W wyniku takich badań[289] opracowano sieci zdolne generować obrazy pisma ręcznego o niewielkiej rozdzielczości ale o dużym zakresie zmienności *inter* i *intra* osobowej (rys. 8.1.1).



Rysunek 8.1.1. Przykład pisma prawdziwego (*Real*) i sztucznie wygenerowanego (*Generated*) za pomocą sieci generatywno-adwersaryjnej, która zdolna jest nauczyć się nawyku pisarskiego danej osoby i generować tekst o dowolnej treści (przy czym, rezultatów tych nie poddano ekspertyzie pismoznawczej). Źródło: L. Kang, P. Riba, Y. Wang, M. Rusiñol, A. Fornés, M. Villegas, *GANwriting: Content-Conditioned Generation of Styled Handwritten Word Image*s, 21 lipca 2020 r., http://arxiv.org/abs/2003.02567.

289 B. Davis, C. Tensmeyer, B. Price, C. Wigington, B. Morse, R. Jain, *Text and Style Conditioned GAN for Generation of Offline Handwriting Lines*, 1 września 2020 r., http://arxiv.org/abs/2009.00678; L. Kang, P. Riba, Y. Wang, M. Rusiñol, A. Fornés, M. Villegas, *GANwriting: Content-Conditioned Generation of Styled Handwritten Word Images*, 21 lipca 2020 r., http://arxiv.org/abs/2003.02567.

Podczas gdy, niewielu badaczy trudni się problematyką generowania podpisów za pomocą sieci neuronowych, ponieważ syntetyzacja podpisów innymi metodami jest szerszą, bardziej popularną i owocną dziedziną[290]. Znane autorowi badania polegały na generowaniu nowych wariantów podpisów na podstawie istniejących podpisów danej osoby, gdzie rezultatem były podpisy o niewielkiej rozdzielczości (256 x 256 px) i zmienności *intra* osobowej[291]. W większości tych przypadków sieć neuronowa, mając wygenerować nowy podpis danej osoby, uczyła się kopiować jeden ze znanych podpisów tej osoby (rys. 8.1.2).

Rysunek 8.1.2. Przykład niskiej zmienności podpisów sztucznych, spowodowanej kopiowaniem przez sieć neuronową jednego z podpisów naturalnych (trzeci od prawej).
Źródło: A.V. Barros da Silva, *Data Augmentation for Offline Handwritten Signature Verification*, Recife 2018.

Stosując standardowe architektury sieci generatywno-adwersaryjnych[292], także celem wytworzenia nowych podpisów na podstawie znanych podpisów danej osoby, autor również napotkał problem niskiej zmienności *intra* osobowej generowanych podpisów. Ogólny problem niskiej zmienności danych uzyskiwanych przez generator sieci określany jest jako *mode collapse*[293], *e.g.* gdy generator osiąga minimum błędu generując jeden lub kilka podobnych obrazów, zaś dyskryminator osiąga minimum

290 M. Diaz, M. Ferrer, G. Ekladious, R. Sabourin, Generation of Duplicated Off-Line Signature Images for Verification Systems, „IEEE Transactions on Pattern Analysis and Machine Intelligence" t. 39 (2016), DOI: 10.1109/TPAMI.2016.2560810; M. Diaz, M. Ferrer, R. Sabourin, Approaching the Intra-Class Variability in Multi-Script Static Signature Evaluation [w:] 2016 23rd International Conference on Pattern Recognition (ICPR), 2016; M.A. Ferrer, M. Diaz-Cabrera, A. Morales, Static Signature Synthesis: A Neuromotor Inspired Approach for Biometrics, „IEEE Transactions on Pattern Analysis and Machine Intelligence" t. 37 nr 3 (2015), DOI: 10.1109/TPAMI.2014.2343981.

291 A.V. Barros da Silva, *Data Augmentation for Offline Handwritten Signature Verification*, Recife 2018.

292 A. Brock, J. Donahue, K. Simonyan, *Large Scale GAN Training for High Fidelity Natural Image Synthesis*, 25 lutego 2019 r., http://arxiv.org/abs/1809.11096.

293 H. Thanh-Tung, T. Tran, *On Catastrophic Forgetting and Mode Collapse in Generative Adversarial Networks*, arXiv, 21 marca 2020 r., http://arxiv.org/abs/1807.04015; L. Weng, *From GAN to WGAN*, arXiv, 18 kwietnia 2019 r., http://arxiv.org/abs/1904.08994.

skupiając się na wykrywaniu tych *stricte* przypadków, a obydwa pozostają we względnej równowadze. Istnieją różne techniki przeciwdziałania temu problemowi[294], ale prowadzić mogą w niektórych przypadkach do spowolnienia uczenia lub obniżenia jakości generowanych danych, ostatecznie nie usuwając problemu. Podczas gdy, techniki podnoszące jakość generowanych danych prowadzić mogą niekiedy do wcześniejszej *mode collapse*. Jednym ze skuteczniejszych sposobów przeciwdziałania *mode collapse* jest zastosowanie wassersteinowskiej funkcji kosztu (*wasserstein loss function*)[295]. Natomiast, podstawową strategią uczenia sieci jest dążenie do podnoszenia jakości generowanych przez nią danych, wraz z doraźnym zapobieganiem *mode collapse* (*i.e.* skoro nie możemy mu permanentnie zapobiec, to staramy się odsunąć je w czasie, aż do uzyskania oczekiwanych rezultatów)[296].

Dlatego, autor postanowił zaprezentować przykład sieci generatywno-adwersaryjnych z którymi wchodzić można w interakcję celem wywierania bezpośredniego wpływu na generowane dane[297]. Zaletą takiej metody jest to, że: i) generować można obrazy w wysokiej rozdzielczości; ii) zakres zmienności generowanych obrazów jest pod kontrolą użytkownika. Przyjęto interakcje z modelem na poziomie danych wejściowych, które można modyfikować lub wprowadzać nowe.

8.2. Metody

Dane. Zbiór skanów anonimowych podpisów pozyskano z bazy CEDAR-Signatures autorstwa CEDAR (*Center of Excellence for Document Analysis and Recognition, State University of New York at Buffalo*)[298]. Były to statyczne obrazy podpisów, wykonanych przez 55 probantów, gdzie każdy udzielił 24 próbek (rys. 8.2.1).

294 A. Brock, J. Donahue, K. Simonyan, *Large Scale GAN Training for High Fidelity Natural Image Synthesis*, 25 lutego 2019 r., http://arxiv.org/abs/1809.11096; H. Thanh-Tung, T. Tran, *On Catastrophic Forgetting and Mode Collapse in Generative Adversarial Networks*, arXiv, 21 marca 2020 r., http://arxiv.org/abs/1807.04015; L. Weng, *From GAN to WGAN*, arXiv, 18 kwietnia 2019 r., http://arxiv.org/abs/1904.08994.

295 L. Weng, *From GAN to WGAN*, arXiv, 18 kwietnia 2019 r., http://arxiv.org/abs/1904.08994.

296 A. Brock, J. Donahue, K. Simonyan, *Large Scale GAN Training for High Fidelity Natural Image Synthesis*, 25 lutego 2019 r., http://arxiv.org/abs/1809.11096.

297 T. Park, M.-Y. Liu, T.-C. Wang, J.-Y. Zhu, *Semantic Image Synthesis with Spatially-Adaptive Normalization*, arXiv, 5 listopada 2019 r., http://arxiv.org/abs/1903.07291.

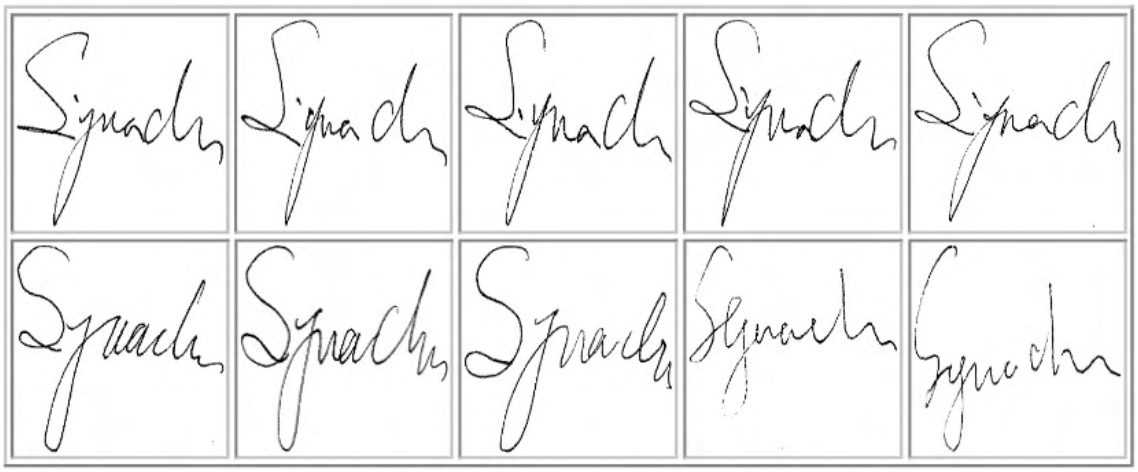
298 S. Dey, A. Dutta, J.I. Toledo, S.K. Ghosh, J. Llados, U. Pal, *SigNet: Convolutional Siamese Network for Writer Independent Offline Signature Verification*, „arXiv:1707.02131 [cs]" (2017), http://arxiv.org/abs/1707.02131; M.K. Kalera, S. Srihari, A. Xu, *Offline signature verification and identification using distance statistics*, „International Journal of Pattern Recognition and Artificial Intelligence" t. 18 nr 07 (2004), DOI: 10.1142/S0218001404003630.

Skany wykonywano w standardzie 300 dpi i zapisywano w skali szarości, stosując format PNG. Probanci nanosili podpisy w okienkach o wymiarach 5.08 x 5.08 cm (2 x 2”). Podpisy nie były pobierane od probantów jednorazowo, ale podczas trzech sesji odbywających się w trzy różne dni.

Rysunek 8.2.1. Przykładowe podpisy wszystkich probantów (CEDAR-Signatures).

Źródło: M.K. Kalera, S. Srihari, A. Xu, *Offline signature verification and identification using distance statistics*, „International Journal of Pattern Recognition and Artificial Intelligence” t. 18 nr 07 (2004), DOI: 10.1142/S0218001404003630.

Pobrano również próbki fałszerstw powyższych podpisów, gdzie żaden z 20 fałszerzy nie był dawcą podpisów oryginalnych. Uzyskano po 24 fałszerstwa na każdego probanta, a więc przypadało po jednym fałszerstwie na oryginał (rys. 8.2.2).



Rysunek 8.2.2. Przykładowe podpisy oryginalne (rząd górny) i sfałszowane (rząd dolny).
Źródło: M.K. Kalera, S. Srihari, A. Xu, *Offline signature verification and identification using distance statistics*, „International Journal of Pattern Recognition and Artificial Intelligence" t. 18 nr 07 (2004), DOI: 10.1142/S0218001404003630.

Należy tutaj odnotować, że w badaniach wstępnych autor ustalił negatywny wpływ podpisów fałszywych na podpisy generowane przez sieć neuronową, co zgodne jest z obserwacjami innych autorów[299]. Oczekiwano, że jeżeli dyskryminator uczony będzie odróżniać podpisy sztuczne i fałszywe od prawdziwych, to generator nauczy się generować podpisy różne od fałszywych, a podobne do prawdziwych. W rezultacie, okazało się że generator nie jest zdolny upodobnić podpisów sztucznych do prawdziwych, zapewniając jednocześnie, że będą one niepodobne do fałszywych.

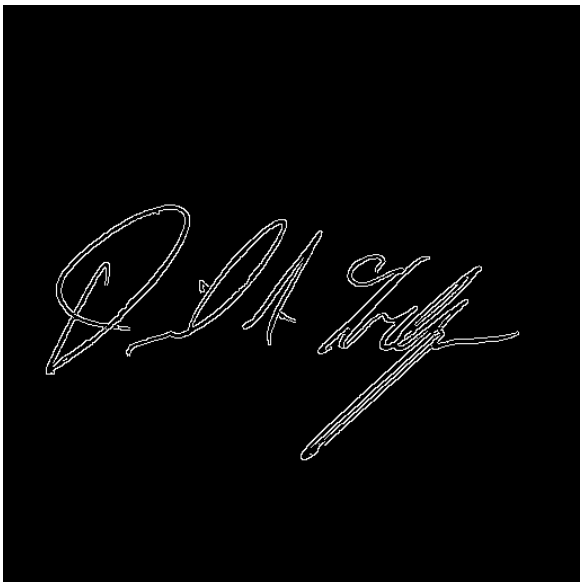
Preprocesowanie. Obrazy podpisów utrzymywane były w skali szarości, przeprowadzano inwersję kolorów, następnie sprawdzano czy obraz ma tą samą wysokość co szerokość, jeżeli nie, to uzupełniano go czarnym tłem (wartość zero), aby przeskalować obrazy do wymiarów 512 x 512 px unikając ich deformacji. Na koniec, obrazy były odszumiane poprzez progowanie wartości pikseli niższych niż 25 do 0.

299 A.V. Barros da Silva, *Data Augmentation for Offline Handwritten Signature Verification*, Recife 2018.

Obrazy, które stanowiły dane wejściowe do dyskryminatora (rys. 8.2.3), były preprocesowane jak opisano powyżej.

Rysunek 8.2.3. Przykład obrazu preprocesowanego, który stanowił dane wejściowe do dyskryminatora.
Źródło: opracowanie własne.

Obrazy, które stanowiły treningowe dane wejściowe do generatora (rys. 8.2.4), były preprocesowane jak opisano powyżej, a następnie rozmywane filtrem gaussowskim, celem wykrywania krawędzi metodą Canny'ego.



Rysunek 8.2.4. Przykład obrazu preprocesowanego, który stanowił treningowe dane wejściowe do generatora.
Źródło: opracowanie własne.

Obrazy, które stanowiły testowe dane wejściowe do generatora (rys. 8.2.5), były tożsame z obrazami treningowymi, ale poddawano je następującym modyfikacjom (augmentacjom): i) losowa elastyczna deformacja obrazu[300]; ii) losowe wycięcie fragmentu obrazu o wymiarach 128 x 128 px, zastępując je czarnym tłem.

Rysunek 8.2.5. Przykład obrazu preprocesowanego, który stanowił testowe dane wejściowe do generatora.

Źródło: opracowanie własne.

Rysunek 8.2.6. Przykład obrazu preprocesowanego, który stanowił maskę generatora, służącą do normalizacji jego wewnętrznych sygnałów.

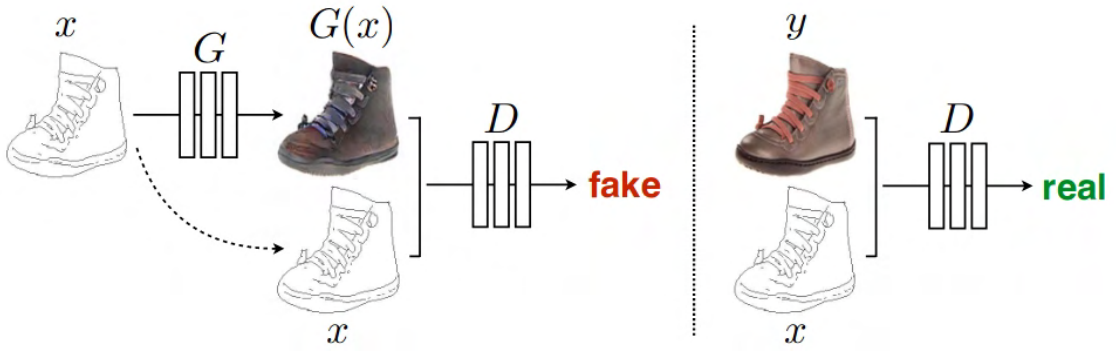
Źródło: opracowanie własne.

300 G. van Tulder, *Elastic deformations for N-dimensional images (Python, SciPy, NumPy, TensorFlow, PyTorch)* [na:] https://pypi.org/project/elasticdeform/, dostęp 27 września 2022 r.

Obrazy, które służyły do normalizacji sygnałów wewnątrz generatora (określane w przypadku modeli translacyjncyh maskami), stanowiły średnie z obrazów podpisów, wyliczane dla każdego probanta osobno (rys. 8.2.6).

Ostatecznie, do uczenia sieci generatywno-adwersaryjnej wykorzystano 240 obrazów podpisów, pochodzących od pierwszych 10 probantów.

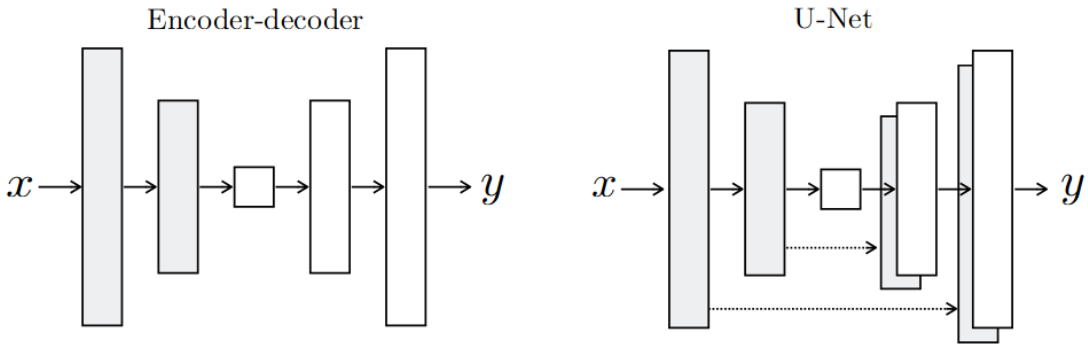
Model. Jako podstawową sieć generatywno-adwersaryjną zastosowano model pix2pix[301], który jest siecią translacyjną (rys. 8.2.7), ponieważ przekształca on jeden obraz w drugi, zachowując przy tym informacje „semantyczne” w nich zawarte (*i.e.* klasy obiektów na obrazach)[302]. Gdzie: i) generator dostaje na wejście szkic prawdziwego obrazu i oddaje na wyjściu obraz sztuczny; ii) dyskryminator dostaje na wejście sztuczny obraz z generatora i szkic prawdziwego obrazu, a docelowa odpowiedź to fałsz; iii) dyskryminator dostaje na wejście obraz prawdziwy i jego szkic, a docelowa odpowiedź to prawda. Generator ma więc za zadanie przekształcić dane wejściowe w wyjściowe, a dyskryminator ma za zadanie odróżnić dane prawdziwe od wygenerowanych przez dyskryminator. Przy tym generator i dyskryminator są warunkowane ze względu na otrzymywane na wejścia szkice prawdziwych danych.



Rysunek 8.2.7. Schemat modelu translacyjnego pix2pix, gdzie generator oznaczono jako G, dyskryminator jako D, dane wejściowe jako x, dane sztuczne jako $G(x)$, zaś dane prawdziwe jako y.
Źródło: P. Isola, J.-Y. Zhu, T. Zhou, A.A. Efros, *Image-to-Image Translation with Conditional Adversarial Networks*, 26 listopada 2018 r., http://arxiv.org/abs/1611.07004.

301 P. Isola, J.-Y. Zhu, T. Zhou, A.A. Efros, *Image-to-Image Translation with Conditional Adversarial Networks*, 26 listopada 2018 r., http://arxiv.org/abs/1611.07004.
302 A. Garcia-Garcia, S. Orts-Escolano, S. Oprea, V. Villena-Martinez, J. Garcia-Rodriguez, *A Review on Deep Learning Techniques Applied to Semantic Segmentation*, arXiv, 22 kwietnia 2017 r., http://arxiv.org/abs/1704.06857.

Autorzy modelu pix2pix oparli architekturę generatora na sieci neuronowej U-Net[303], która podobna jest do sieci typu *encoder-decoder*, gdzie dane wejściowe i wyjściowe mają tą samą rozdzielczość, przy czym dane wejściowe sieć najpierw zakodowuje, stopniowo zmniejszając ich rozdzielczość, a następnie odkodowuje, stopniowo zwiększając ich rozdzielczość. W przypadku modelu U-Net, ustanawiane są dodatkowe połączenia pomiędzy odpowiadającymi sobie rozdzielczością warstwami encodera i decodera (rys. 8.2.8).

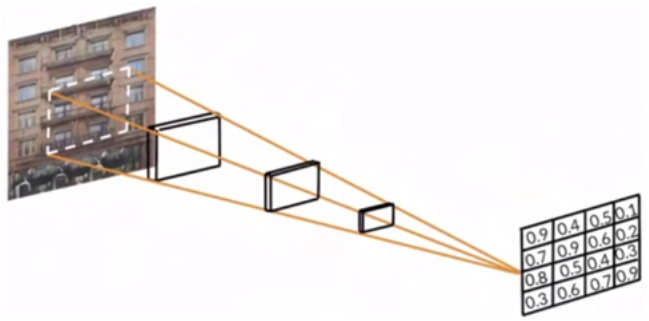


Rysunek 8.2.8. Schemat generatora pix2pix (sieć U-Net).

Źródło: P. Isola, J.-Y. Zhu, T. Zhou, A.A. Efros, *Image-to-Image Translation with Conditional Adversarial Networks*, 26 listopada 2018 r., http://arxiv.org/abs/1611.07004.

Architektura dyskryminatora pix2pix jest stosunkowo prosta (*i.e.* jest to standardowa pięciowartościowa sieć konwolucyjna), ale posiada unikalne rozwiązanie dotyczące wyjścia z sieci. Otóż, w typowym klasyfikatorze lub dyskryminatorze liczba neuronów na wyjściu odpowiada liczbie możliwych odpowiedzi. W przypadku modelu pix2pix, wyjście z dyskryminatora ma wymiary 32 x 32 neurony, gdzie każdy z nich orzeka o prawdziwości lub fałszywości fragmentu obrazu (rys. 8.2.9), stąd określany jest jako *PatchGAN*. W omawianym przypadku, obraz z generatora miał wymiary 512 x 512 px, więc każdy z neuronów wyjściowych dyskryminatora wypowiadał się o prawdziwości fragmentów o wymiarach 16 x 16 px.

303 O. Ronneberger, P. Fischer, T. Brox, *U-Net: Convolutional Networks for Biomedical Image Segmentation*, arXiv, 18 maja 2015 r., http://arxiv.org/abs/1505.04597.



Rysunek 8.2.9. Schemat uproszczony dyskryminatora pix2pix (*PatchGAN*), gdzie dyskryminator klasyfikuje fragmenty obrazu wejściowego indywidualnie.
Źródło: https://velog.io/@tobigs-gm1/Image-to-Image-Translation, dostęp 3 września 2022 r.

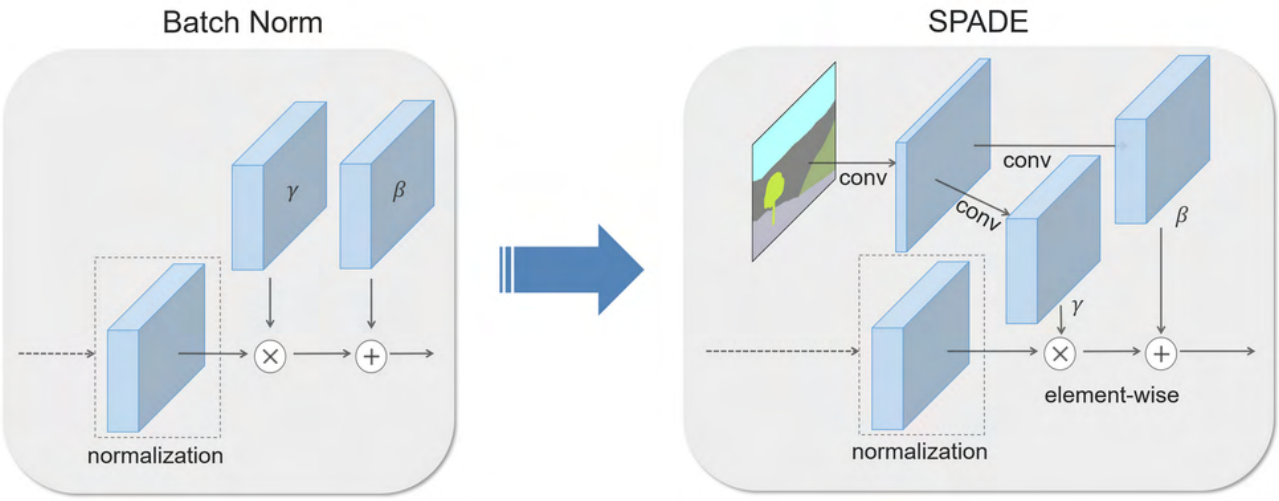
Ponieważ model pix2pix przystosowany był do generowania obrazów o wymiarach 256 x 256 px, a w omawianym eksperymencie przetwarzano obrazy o wymiarach 512 x 512 px, to dodano dwie warstwy do generatora (po jednej warstwie dla encodera i decodera). Dodano też dodatkową warstwę do dyskryminatora, ale znacznie pogarszała ona rezultaty, została więc pominięta.

W modelu pix2pix, oraz w większości sieci generatywno-adwersaryjnych, zastosowanie znajdują warstwy, które normalizują sygnały między warstwami konwolucyjnymi, zapewniając że sygnały te będą mieć średnią bliską 0 i odchylenie standardowe bliskie 1. Normalizacja, która odbywa się dla warstwy konwolucyjnej wobec jej sygnałów wyjściowych uzyskanych z danej liczby danych wejściowych, nazywana jest *batch-normalization* (gdzie *batch-size* odnosi się do liczby danych wejściowych, które sieć musi przetworzyć, zanim zostanie ze względu na nie skorygowana). Okazuje się jednak, że *batch-normalization* znacznie pogarsza rezultaty sieci generatywno-adwersaryjnych[304] (czego autor doświadczył też w swoich badaniach). Jednym z rozwiązań tego problemu jest normalizacja instancyjna (*instance-normalization*), która odbywa się dla neuronów danej warstwy ze względu na ich sygnały uzyskane z jednego egzemplarza danych wejściowych. Rozwiązanie to autor zastosował w dyskryminatorze modelu pix2pix (zamiast *batch-normalization*).

W obydwu powyższych metodach normalizacji, model uczy się dwóch dodatkowych parametrów: i) wektor beta β umożliwiający sieci korektę znormalizowanej średniej; ii) wektor gamma γ umożliwiający sieci korektę znormalizowanego odchylenia standardowego. Jak zauważyli jednak autorzy sieci

304 T. Park, M.-Y. Liu, T.-C. Wang, J.-Y. Zhu, *Semantic Image Synthesis with Spatially-Adaptive Normalization*, arXiv, 5 listopada 2019 r., http://arxiv.org/abs/1903.07291.

neuronowej GauGAN[305], w przypadku modeli translacyjnych, normalizacja powyższymi metodami prowadzi do „wypłukiwania informacji semantycznych” (”*normalization layers tend to “wash away” information contained in the input semantic masks*”[306]). Zaproponowali więc normalizację z wykorzystaniem wyższego stopnia tensorów[307] beta i gamma, która umożliwiać powinna przestrzenną adaptacyjność tych współczynników, pozwalając im na przekazywanie informacji „semantycznych” zawartych w tzw. maskach. Gdzie, maska jest dwuwymiarowym obrazem pełniącym rolę „mapy semantycznej” (*semantic map*), zawierającym wybrane informacje o obrazie docelowym (np. szkic docelowego obrazu). Powyższą metodę normalizacji nazwano przestrzennie-adaptatywną normalizacją (*Spatially-Adaptive Normalization*, SPADE), gdzie: i) sygnały wyjściowe z warstwy konwolucyjnej, a więc jej mapy aktywności, są normalizowane; ii) maska przetwarzana jest przez osobną warstwę konwolucyjną, a następnie przez dwie kolejne warstwy konwolucyjne, po jednej dla tensora gamma i beta, które są w ten sposób uzyskiwane; iii) tensor gamma mnożony jest ze znormalizowanymi mapami aktywności, a tensor beta jest dodawany (rys. 8.2.10).



Rysunek 8.2.10. Schemat *batch-normalization* (*Batch Norm*) i SPADE, gdzie: i) *normalization* to normalizacja; ii) *conv* to konwolucja; iii) iv) tensory beta i gamma oznaczono jako β i γ; iv) iloczyn i sumę tensorów oznaczono jako $\times$ i $+$.

Źródło: https://nvlabs.github.io/SPADE/, dostęp 3 września 2022 r.

305 Ibid.
306 Ibid.
307 Wektor to tensor pierwszego rzędu lub „macierz jednowymiarowa”. W tym przypadku mamy na myśli tensor co najmniej drugiego rzędu lub „macierz co najmniej dwuwymiarową”.

Ponieważ autor dokonał implementacji SPADE modyfikując generator modelu pix2pix, to: i) w przeciwieństwie do modelu GauGAN, danymi wejściowymi do generatora nie były liczby losowane z dystrybucji gaussowskiej (stąd nazwa modelu), ale kontury docelowych obrazów (*vide* szkice obrazów); ii) warstwy SPADE otrzymywały maski w postaci średnich wyciąganych z obrazów danej osoby. W alternatywnych przypadkach, kiedy maskami były: i) sumy obrazów, to rezultaty były bardzo podobne; ii) prawdziwe obrazy, to model uczył się kopiować maski. Warto tutaj odnotować, że podjęto próby, w których dyskryminator otrzymywał nie tylko obraz kwestionowany i obraz wejściowy (kontury podpisu), ale też maskę. W rezultacie dyskryminator uczył się za szybko w stosunku do generatora i nie osiągano zamierzonych wyników.

Jako funkcję kosztu dla generatora i dyskryminatora zastosowano wassersteinowski wariant funkcji *hinge loss*[308], którą miały minimalizować:

$$\begin{aligned} L_G &= -\, D(G(x,m),x) \\ L_D &= \quad max(0, 1 + D(G(x,m),x)) \\ &\quad + max(0, 1 - D(v,x)) \end{aligned} \qquad \text{(Równanie 8.1.1)}$$

Gdzie:

L_G – koszt generatora;

L_D – koszt dyskryminatora;

G – funkcja generator;

D – funkcja dyskryminator;

x – dane wejściowe do generatora;

m – maska dla generatora;

v – dane prawdziwe;

$max(a,b)$ – funkcja wskazująca element o najwyższej wartości z dwóch podanych alternatyw (a,b).

W rezultacie: i) jeżeli dyskryminator oceni obraz z generatora na -1 (fałsz), to koszt minimalizowany przez generator wynosił będzie 1; ii) jeżeli dyskryminator oceni obraz

308 A. Brock, J. Donahue, K. Simonyan, *Large Scale GAN Training for High Fidelity Natural Image Synthesis*, 25 lutego 2019 r., http://arxiv.org/abs/1809.11096; The TensorFlow GAN Authors, *Losses that are useful for training GANs* [na:] https://github.com/tensorflow/gan/blob/master/tensorflow_gan/python/losses/losses_impl.py, dostęp 27 września 2022 r.

z generatora na 1 (prawda), to koszt minimalizowany przez generator wynosił będzie -1; iii) jeżeli dyskryminator oceni obraz z generatora na 0 (niezdecydowany), to koszt minimalizowany przez generator wynosił będzie 0. W przypadku kosztu dyskryminatora, jest on sumą dwóch pomniejszych kosztów, gdzie: i) wobec obrazu z generatora $G(x, m)$ i obrazu wejściowego do generatora x, oczekiwana odpowiedź dyskryminatora to -1 lub mniej; ii) wobec obrazu prawdziwego v i obrazu wejściowego do generatora x, oczekiwana odpowiedź dyskryminatora to 1 lub więcej.

Pod względem parametrów, zastosowano: i) optymizator Adam (*Adaptive Moment Estimation*)[309] z parametrami rekomendowanymi przez autorów modelu pix2pix[310]; *batch-size* wynosił 1; trening trwał 260 epok.

Interaktywność. Podstawową zaletą modeli translacyjnych jest możliwość warunkowania danych wyjściowych na podstawie danych wejściowych. Stąd modyfikacje danych wejściowych (które nie wykraczają poza to, czego nauczył się model), umożliwiają generowanie nowych danych wyjściowych wedle intencji użytkownika (który dane wejściowe wcześniej zmodyfikował). W rezultacie, zakresy zmienności danych, przede wszystkim zakresy zmienności *inter* i *intra* osobowej, znajdują się pod zupełną kontrolą użytkownika.

Jako uczące dane wejściowe zastosowano kontury podpisów, ponieważ opracowanie konturu fałszywego podpisu powinno być dla człowieka łatwiejsze, niż sfałszowanie podpisu w ogóle. Szczególnie, jeżeli dysponuje on większą liczbą podpisów oryginalnych i wiadomościami pismoznawczymi. Ponieważ, uzupełnienie i wypełnienie konturów jest już zadaniem sieci neuronowej. W hipotetycznym przypadku, że sieć generatywno-adwersaryjna potrafi generować sztuczne podpisy w sposób niewykrywalny (*e.g.* wykonuje ona też odpowiednie tło podpisu), problem wykrycia fałszerstwa sprowadza się do ujawnienia cech dyskryminatywnych, wynikających z błędów człowieka, który opracował kontur podpisu.

W celu zapewnienia obiektywnych warunków eksperymentu, kontury fałszywych podpisów nie były wykonywane przez autora, ale przez program, który losowo zniekształcał kontury prawdziwych podpisów. Losowe usuwanie fragmentów

309 D.P. Kingma, J. Ba, *Adam: A Method for Stochastic Optimization*, arXiv, 29 stycznia 2017 r., http://arxiv.org/abs/1412.6980.

310 P. Isola, J.-Y. Zhu, T. Zhou, A.A. Efros, *Image-to-Image Translation with Conditional Adversarial Networks*, 26 listopada 2018 r., http://arxiv.org/abs/1611.07004.

konturów służyło upewnieniu się, że model nie będzie rekonstruował ich na podstawie zapamiętanych podpisów, a będzie je pomijał (służyło to wykluczeniu *mode collapse*).

8.3. Rezultaty i dyskusja. W wyniku przeprowadzonych badań udało się opracować generatywno-adwersaryjną sieć neuronową do fałszowania podpisów, która osiągnęła wstępnie satysfakcjonujące rezultaty.

Trening modelu trwał 260 epok i został przerwany w związku z brakiem postępów w uczeniu się modelu (tab. 8.3.1).

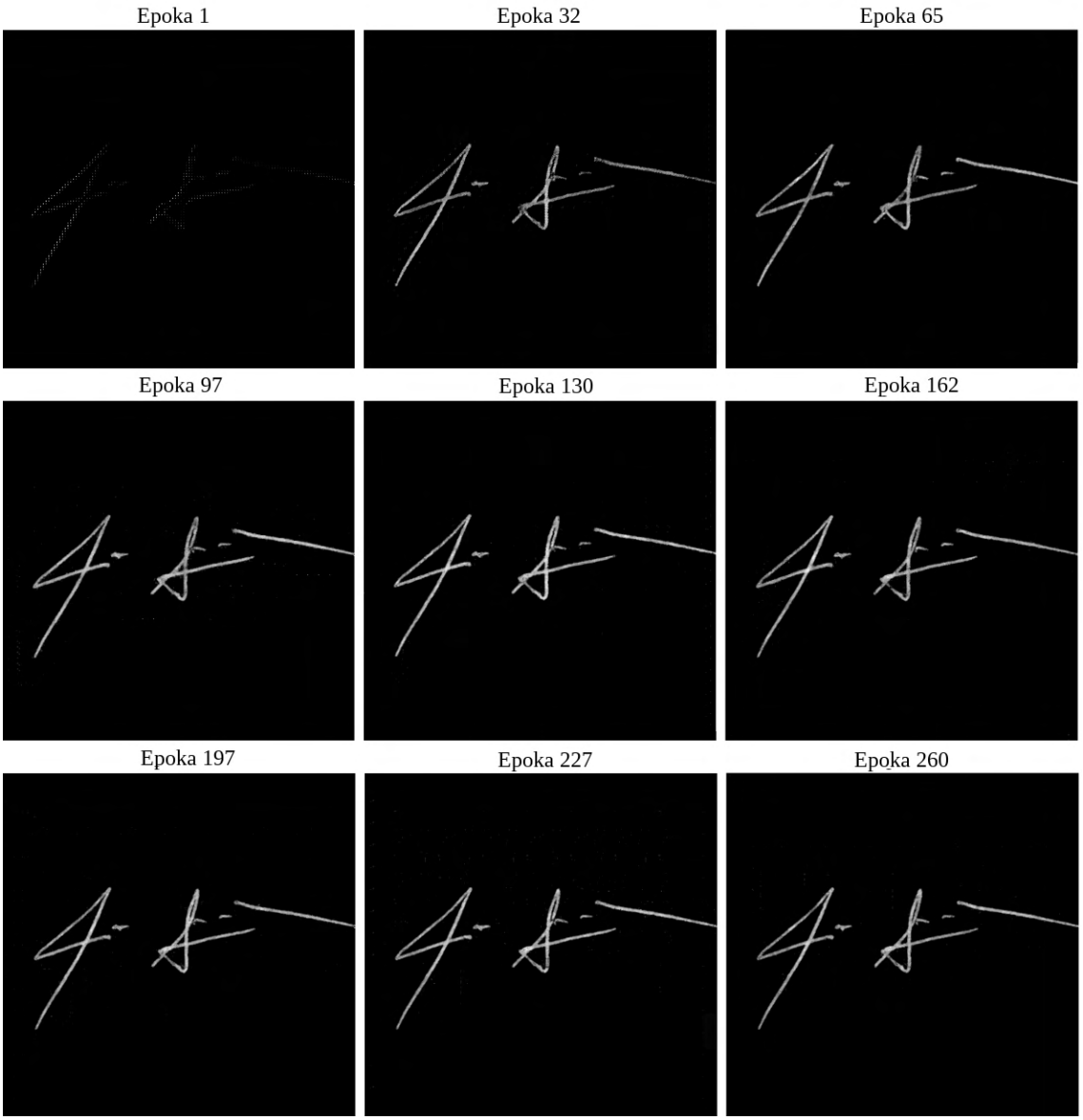
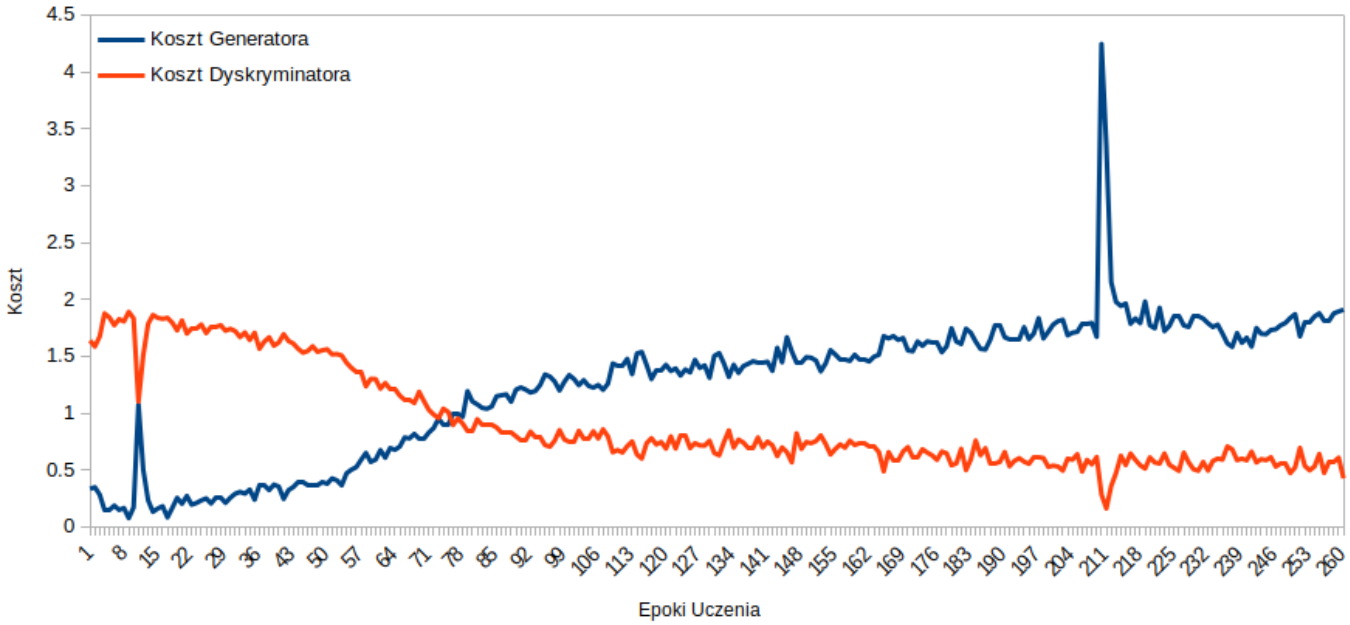


Tabela 8.3.1. Przykład postępów modelu w kolejnych epokach.

Źródło: opracowanie własne.

Warto przy tym zauważyć, że trening modelu był stabilny i udało się osiągnąć równowagę (zbieżność)[311] pomiędzy kosztem generatora i dyskryminatora (rys. 8.3.1).



Rysunek 8.3.1. Wykres kosztu generatora i dyskryminatora podczas treningu.

Źródło: opracowanie własne.

Ponieważ decyzja o tym, w której epoce uzyskano najlepsze rezultaty, jest w przypadku modeli generatywno-adwersaryjnych wysoce arbitralna, to autor postanowił zaprezentować i przeanalizować rezultaty uzyskane w ostatniej epoce uczenia, *i.e.* 260. Warto tutaj odnotować, że standardowymi metodami ewaluacji sztucznych obrazów są *Inception Score* (IS) i *Fréchet Inception Distance* (FID)[312], które polegają na sieci neuronowej do klasyfikacji obrazów *InceptionNet*, mierząc różnicę między pobudzeniami jakie wywołują w tej sieci obrazy naturalne a sztuczne. Ponieważ sieć ta uczona była na zbiorze obrazów *ImageNet*, który nie zawiera klas pismoznawczych, więc metody te nie mogą być tutaj przydatne.

Jak można zaobserwować (tab. 8.3.2), podczas uczenia modelu udało się uniknąć *mode-collapse*, gdyż generuje on podpisy sztuczne na podstawie konturów zmodyfikowanych podpisów oryginalnych. Ponadto, model nie uzupełnia losowo

311 S. Sidheekh, A. Aimen, N.C. Krishnan, *On Characterizing GAN Convergence Through Proximal Duality Gap* [w:] *Proceedings of the 38th International Conference on Machine Learning*, PMLR 2021.

312 A. Borji, *Pros and Cons of GAN Evaluation Measures: New Developments*, 2 października 2021 r., http://arxiv.org/abs/2103.09396.

usuwanych fragmentów podpisów, a to również wskazywałoby na przeuczenie do jednego wzorca, gdyby model rekonstruował nieznane fragmenty.

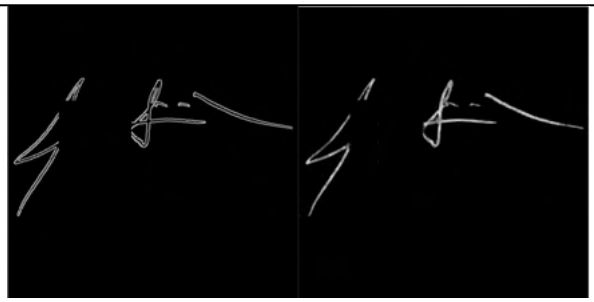
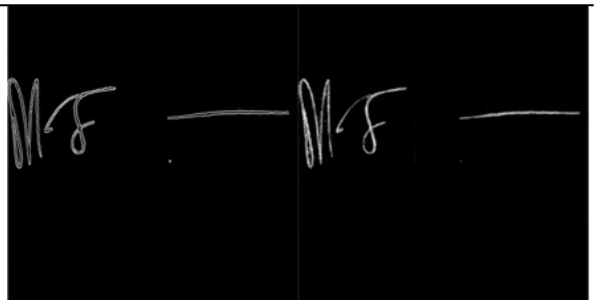
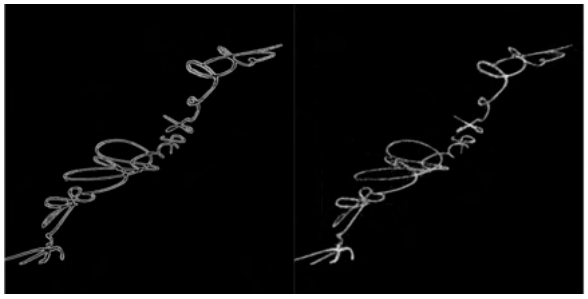
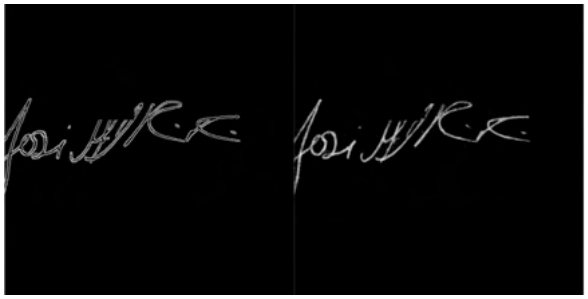
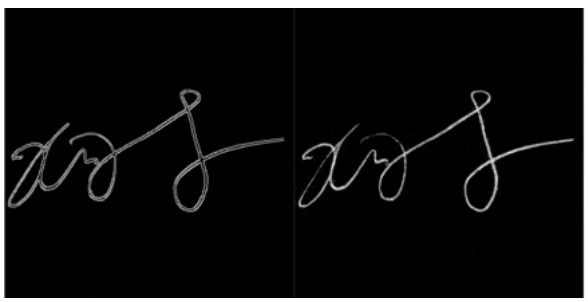
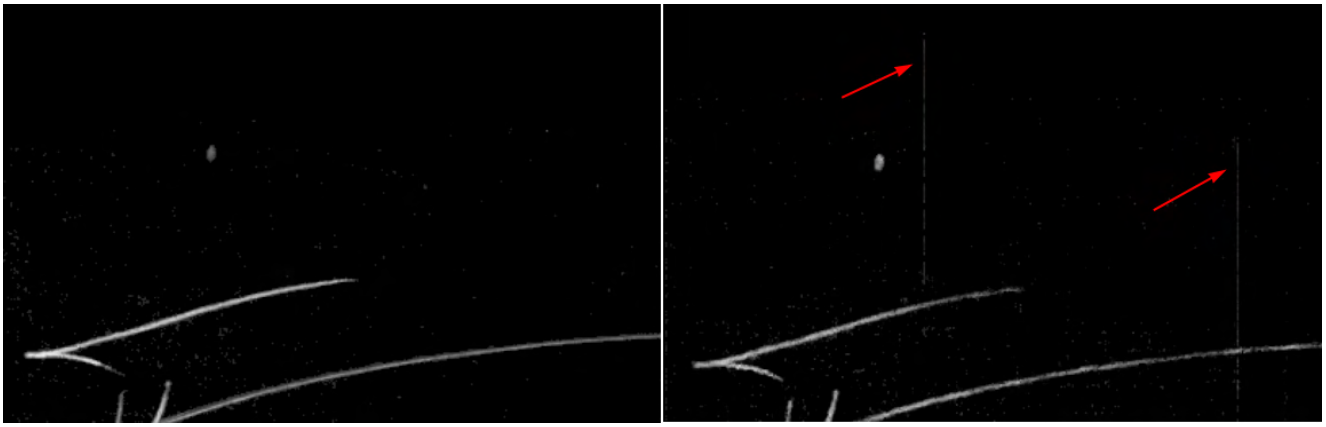
Identyfikator Probanta	Zmodyfikowany Kontur	Wygenerowany Obraz	Identyfikator Probanta	Zmodyfikowany Kontur	Wygenerowany Obraz
1			6		
2			7		
3			8		
4			9		
5			10		

Tabela 8.3.2. Przykładowe obrazy testowe wygenerowane przez model w 260 epoce uczenia.

Źródło: opracowanie własne.

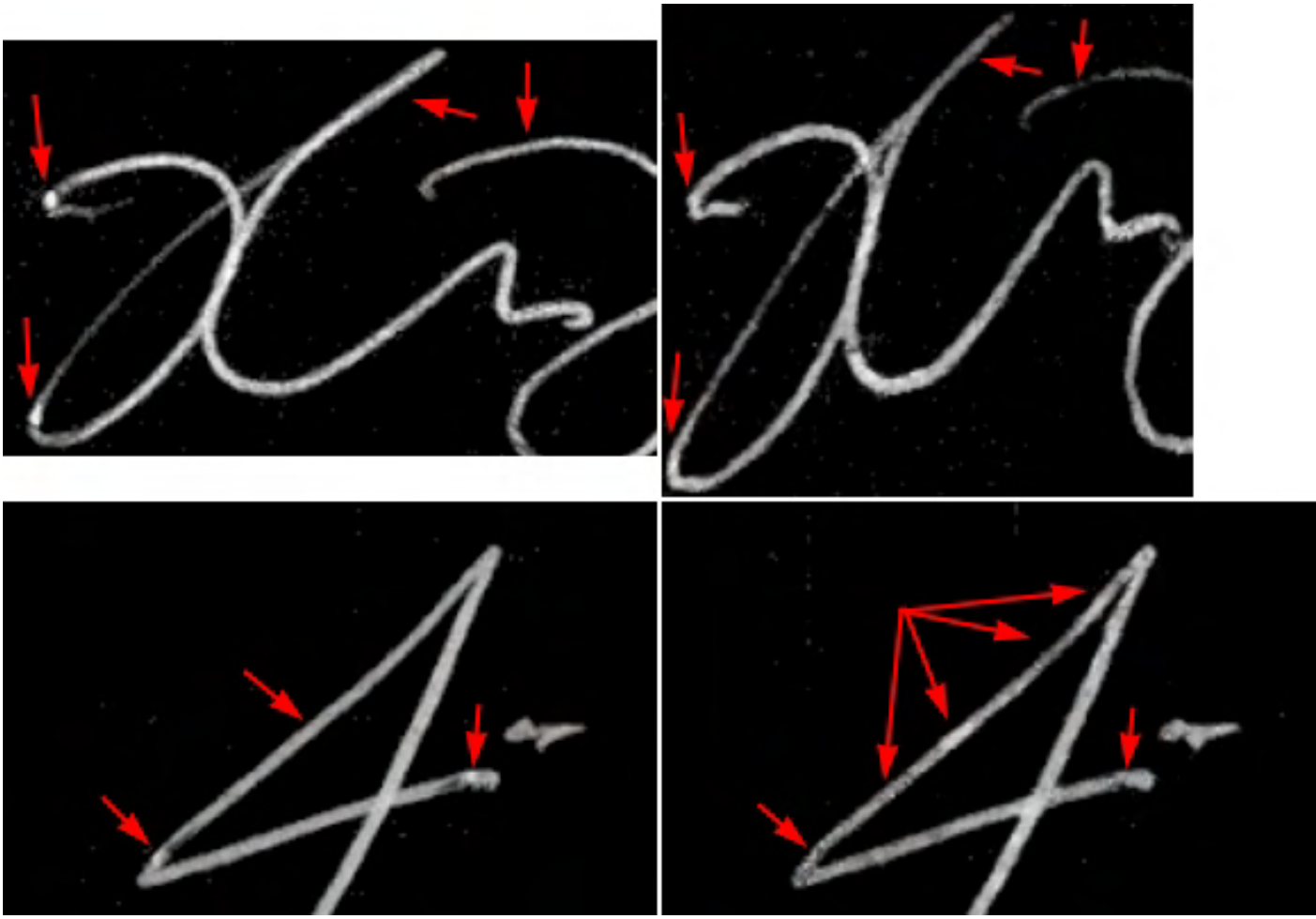
Jednakże, jakość generowanych podpisów nie spełnia oczekiwań, bowiem zawierają one wiele cech umożliwiających łatwą ich dyskryminację jako fałszywych.

Po pierwsze, obrazy sztuczne zawierają odmienne ilości szumu, niż były obecne w obrazach oryginalnych. Ponadto, szum ten jest odmiennej jakości niż na obrazach prawdziwych. W poniższym przykładzie (rys. 8.3.2), model naniósł dwie krótkie linie wertykalne, które pojawiają się na niektórych obrazach oryginalnych z powodu niedoskonałości skanera użytego przez autorów bazy danych. Jednakże, są to na ogół linie ciągłe (linia znajdująca się po lewej obrazu sztucznego jest wyraźnie przerywana), biegną przez całą długość obrazu (linie widoczne na obrazie sztucznym biegną tylko przez jego fragment), oraz w obrazach prawdziwych występuje tylko jedna taka linia na skan (na przykładzie obrazu sztucznego widoczne są dwie).



Rysunek 8.3.2. Przykład różnic w ilości i jakości szumu na obrazie oryginalnym (po lewej) i sztucznym (po prawej). Przykładowe różnice oznaczono czerwonymi strzałkami.
Źródło: opracowanie własne.

Po drugie, występują istotne różnice dotyczące ilości i jakości cieniowania na obrazach sztucznych. Na poniższym przykładzie (rys. 8.3.3), poszczególne grammy w obrazach naturalnych są na ogół równomiernie cieniowane, a zmiany ich cieniowania następują płynnie. Natomiast, w przypadku obrazów sztucznych, cieniowanie na grammach znacząco odbiega od podpisów naturalnych, ponieważ jest ono nierównomierne, a jego zmiany następują gwałtownie. Ponadto, pomiędzy podpisami naturalnymi i sztucznymi występują istotne różnice w częstotliwości, kierunku i miejscach wzmożenia nacisku. Istotnie, punkty najbardziej wzmożonego nacisku występujące w podpisach naturalnych, nie występują w podpisach sztucznych.



Rysunek 8.3.3. Przykład różnic w ilości i jakości cieniowania na obrazach oryginalnych (po lewej) i sztucznych (po prawej). Podpis sztuczny u góry (po prawej), powstał na podstawie zmodyfikowanego konturu podpisu oryginalnego (po lewej). Natomiast, podpis sztuczny u dołu (po prawej), powstał na podstawie niemodyfikowanego konturu podpisu oryginalnego (po lewej). Przykładowe różnice oznaczono czerwonymi strzałkami.

Źródło: opracowanie własne.

Po trzecie, występują istotne różnice ilościowe i jakościowe linii na obrazach oryginalnych i sztucznych (rys. 8.3.4). Pod względem jakości, w przypadku obrazów naturalnych kontury linii są równe i płynne, podczas gdy na obrazach sztucznych kontury linii charakteryzują się nienaturalnymi ząbkowaniami, wybrzuszeniami, ubytkami i falowaniem. Pod względem ilościowym, zmiany grubości linii na obrazach oryginalnych następują płynnie, podczas gdy na obrazach sztucznych grubość linii ulega gwałtownym zmianom.

Rysunek 8.3.4. Przykład różnic ilościowych i jakościowych na obrazach oryginalnych (po lewej) i sztucznych (po prawej). Brakujące znaki na obrazie sztucznym u dołu (po lewej), wynikają z modyfikacji jakie przeprowadzono wobec konturu obrazu prawdziwego, gdzie losowo usunięto brakujący fragment. Przykładowe różnice oznaczono czerwonymi strzałkami.
Źródło: opracowanie własne.

Po czwarte, opracowany model generatywno-adwersaryjny nie radzi sobie z generowaniem obrazów, na których występują blisko położone linie równoległe lub prostopadłe, oraz pętlice o niewielkiej średnicy (rys. 8.3.5). W sytuacjach tych dochodzi bowiem do zlewania się linii na obrazach sztucznych.

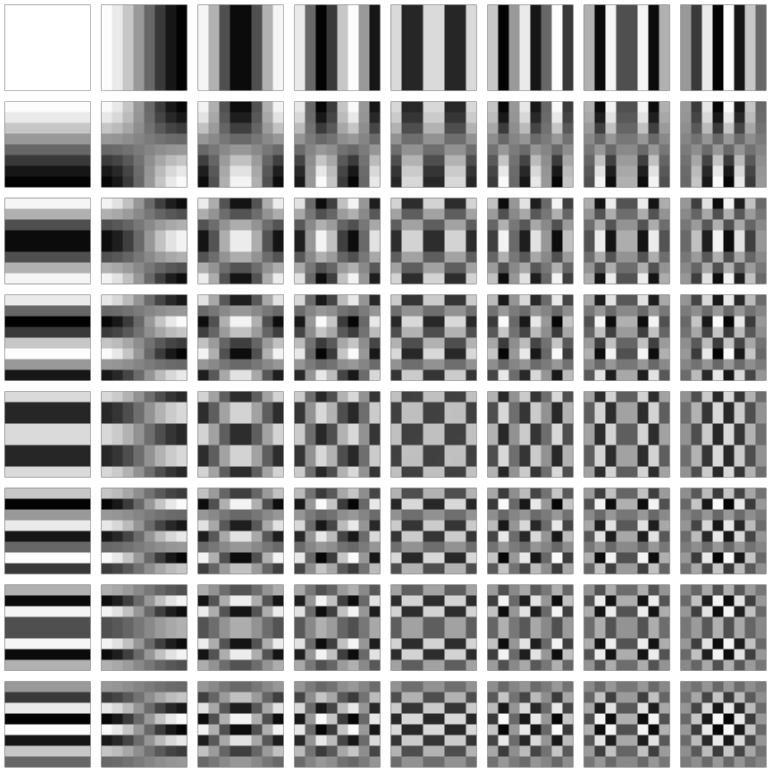
Najbardziej prawdopodobną przyczyną tego problemu są dane wejściowe w postaci konturów podpisów, których niedoskonałości wprowadzać mogą model w błąd, oraz które mogą nie zawierać wszystkich relewantnych informacji o przebiegu linii znaków. Można stąd przypuszczać, że lepszym rozwiązaniem byłyby dane wejściowe w postaci binaryzowanych podpisów, a nie ich konturów. Skoro w przypadku konturów zadaniem modelu jest identyfikacja konturów i odpowiednie wypełnienie ich teksturą o właściwym cieniowaniu (*i.e.* pikselami o właściwych wartościach). To w przypadku binaryzowanych obrazów podpisów zadaniem modelu będzie tylko modyfikacja

wartości pikseli innych niż zero (gdzie piksele tła mają wartość zero). Szczególnie, że dla człowieka prostszym zadaniem będzie narysowanie lub zmodyfikowanie istniejącego podpisu w jego zbinaryzowanej postaci, niż uczynienie tego samego dla konturu podpisu (*e.g.* ekstrakcja konturu po modyfikacji zbinaryzowanego podpisu).

Rysunek 8.3.5. Przykład zlewania się linii na obrazach sztucznych. Obrazy prawdziwe znajdują się po lewej, obrazy sztuczne znajdują się po prawej, a relewantne kontury znajdują się pod tymi obrazami. Przykładowe różnice oznaczono czerwonymi strzałkami.

Źródło: opracowanie własne.

Ostatecznie, autor postanowił sprawdzić możliwość wykrywania obrazów sztucznych – pochodzących z opracowanej sieci generatywno-adwersaryjnej – za pomocą jednej z najbardziej uniwersalnych metod, którą jest analiza widma częstotliwościowego sygnału (*i.e.* obrazu), ujawniająca artefakty świadczące o sztuczności obrazu[313]. W tym celu zastosowano dyskretną transformatę kosinusową (*Discrete Cosine Transform*, DCT), która zakłada, że obraz jest sumą funkcji kosinusowych o różnych częstotliwościach (rys. 8.3.6), wyważonych przez współczynniki określające ich udział w oryginalnym obrazie. Wartość tych współczynników jest wizualizowana (rys. 8.3.7), gdzie lewy górny róg odpowiada niższym częstotliwościom, a prawy dolny róg odpowiada wyższym częstotliwościom funkcji kosinusowych. Ponieważ niższe częstotliwości mają największy wpływ na treść obrazu, to możemy zaobserwować dla nich współczynniki o najwyższych wartościach.

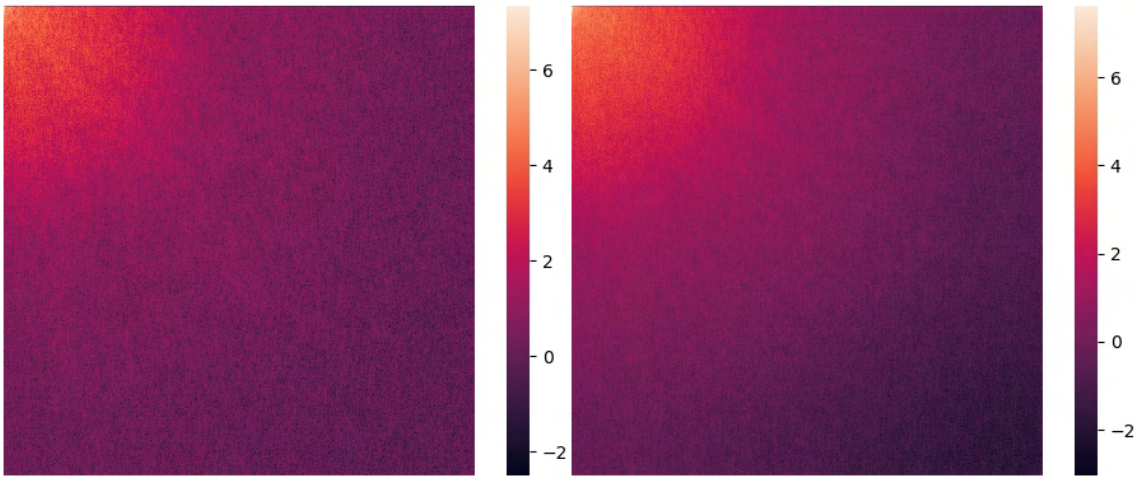


Rysunek 8.3.6. Przykład uproszczony matrycy dwuwymiarowych funkcji kosinusowych o różnych częstotliwościach. Gdzie, lewy górny róg odpowiada niższym częstotliwościom, a prawy dolny róg odpowiada wyższym częstotliwościom funkcji kosinusowych

Źródło: https://upload.wikimedia.org/wikipedia/commons/2/24/DCT-8x8.png, dostęp 8 września 2022 r.

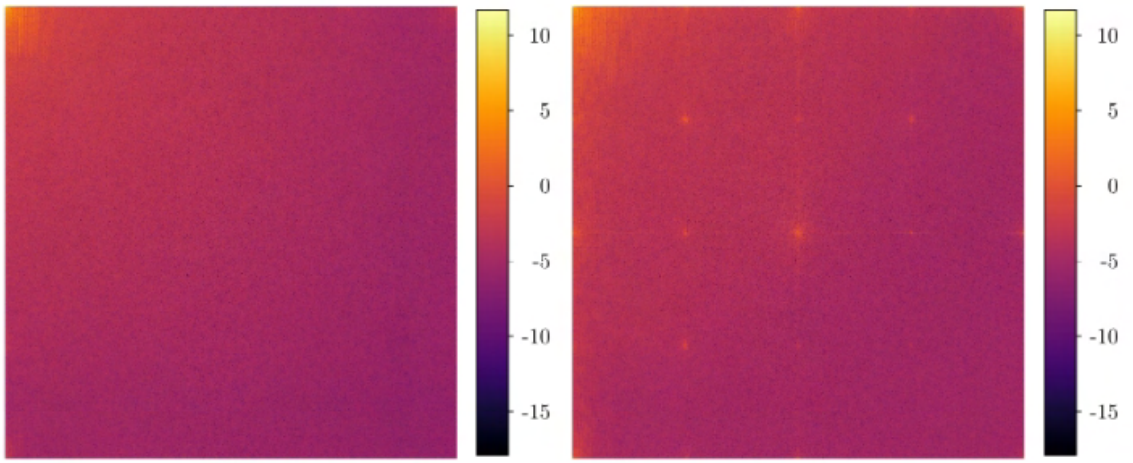
313 J. Frank, T. Eisenhofer, L. Schönherr, A. Fischer, D. Kolossa, T. Holz, *Leveraging Frequency Analysis for Deep Fake Image Recognition*, arXiv, 26 czerwca 2020 r., http://arxiv.org/abs/2003.08685.

Przeprowadzona transformata kosinusowa obrazów sztucznych i prawdziwych została dla nich oddzielnie uśredniona (rys. 8.3.7). W przeciwieństwie do autorów tej metody wykrywania sztuczności obrazów (rys. 8.3.8), nie zaobserwowano artefaktów świadczących bezsprzecznie o wygenerowaniu obrazów przez model (rys. 8.3.7).



Rysunek 8.3.7. Widmo częstotliwościowe obliczone za pomocą dyskretnej transformaty kosinusowej (*Discrete Cosine Transform*, DCT) dla obrazów prawdziwych (po lewej) i sztucznych (po prawej). W obydwu przypadkach są to średnie widma z testowych obrazów prawdziwych i sztucznych. Dla lepszej widoczności zastosowano skalę logarytmiczną.

Źródło: opracowanie własne.



Rysunek 8.3.8. Przykład widma częstotliwościowego obliczonego za pomocą dyskretnej transformaty kosinusowej (*Discrete Cosine Transform*, DCT) dla obrazów prawdziwych (po lewej) i sztucznych (po prawej). Gdzie widmo obrazu sztucznego pozwala zaobserwować artefakty świadczące o jego fałszywości (rozjaśnione punkty i linie przypominające kratę). Model uczony był na bazie danych FFHQ.

Źródło: J. Frank, T. Eisenhofer, L. Schönherr, A. Fischer, D. Kolossa, T. Holz, *Leveraging Frequency Analysis for Deep Fake Image Recognition*, arXiv, 26 czerwca 2020 r., http://arxiv.org/abs/2003.08685.

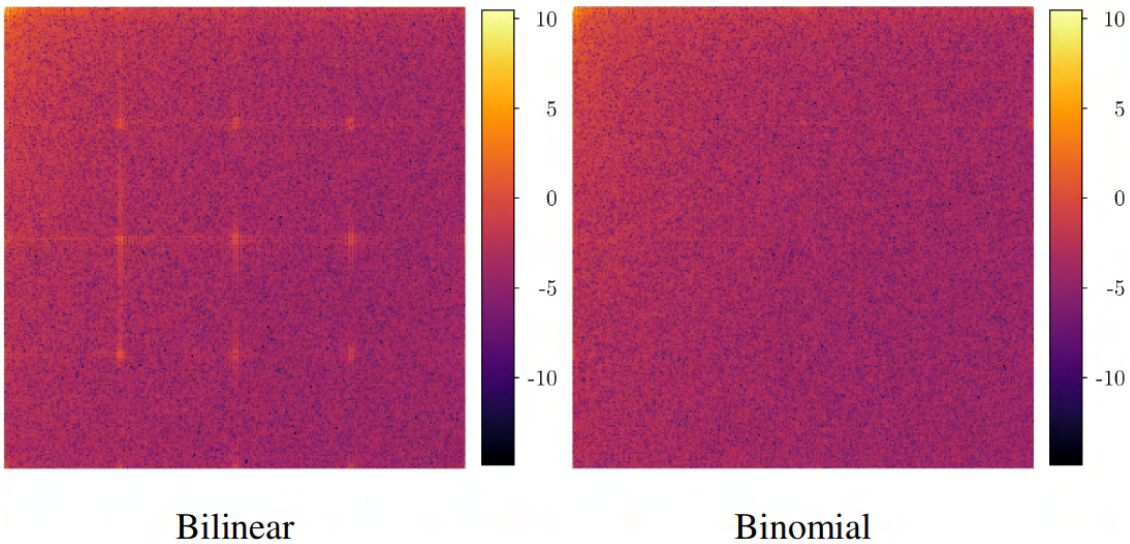
Nie oznacza to jednak, że nie można wskazać żadnych różnic między widmami opracowanych obrazów sztucznych i prawdziwych. Otóż, zaobserwować można dla obrazów sztucznych: i) wyższe wartości współczynników dla niskich częstotliwości; ii) niższe wartości współczynników dla wyższych częstotliwości. Autor nie jest jednak przekonany, iż byłoby to wystarczające do stwierdzenia sztuczności danego obrazu.

Najbardziej prawdopodobnym wyjaśnieniem dla zawodności tej metody w omawianym przypadku, jest powód dla którego artefakty wskazane przez jej autorów są w ogóle obecne w widmach obrazów sztucznych. Otóż, zastosowanie w generatorze transponowanej konwolucji służy zwiększeniu rozdzielczości przetwarzanego obrazu, powodować ono jednak może kratkowane tekstury obrazów (rys. 8.3.9). Stąd, popularnym rozwiązaniem problemu jest zastosowanie w generatorze warstw konwolucyjnych (których parametry nie powinny prowadzić przy tym do zmniejszania rozdzielczości), a podnoszenie rozdzielczości pomiędzy tymi warstwami za pomocą innych metod (warstw zwiększających rozdzielczość, *upsampling layers*), które mogą być jednak przyczyną występowania zaobserwowanych artefaktów w widmie częstotliwościowym obrazu.

Rysunek 8.3.9. Przykład obrazu z sieci generatywno-adwersaryjnej, gdzie pojawiają się kratkowania tekstury (*grid-like patterns* / *checkerboard artifacts*).
Źródło: https://la.mathworks.com/help/examples/nnet/win64/xxexp_mgr_gan_test_images2.png, dostęp 8 września 2022 r.

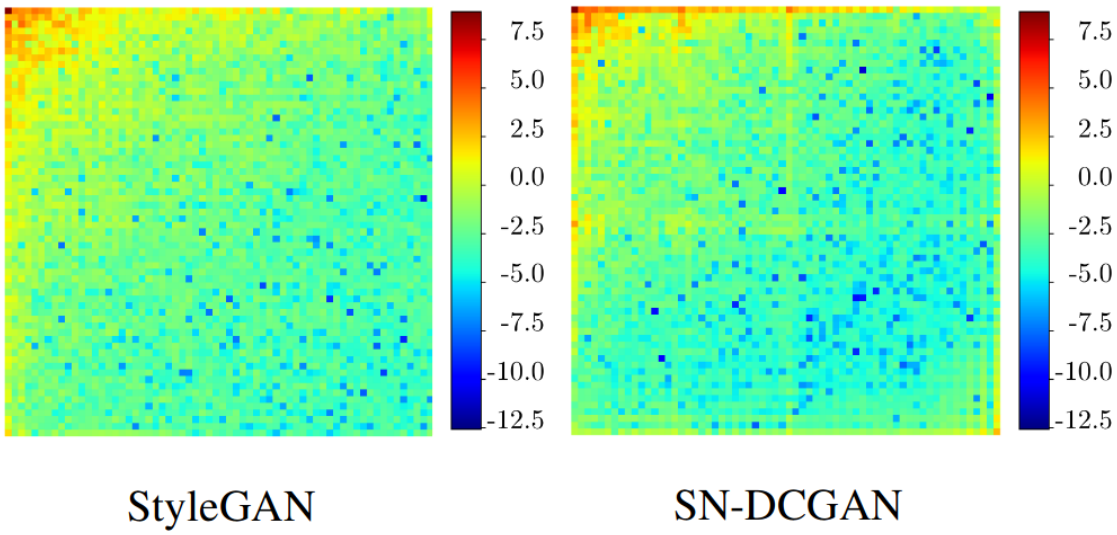
Tak też autorzy omawianej metody postanowili zbadać wpływ różnych metod podnoszenia rozdzielczości na model StyleGAN uczony na bazie danych FFHQ, dowodząc, że od wyboru tych metod zależy obecność obserwowanych artefaktów (rys. 8.3.10). Przy czym, ustalili oni na przykładzie modelu SN-DCGAN, że również w przypadku, gdy za regulację rozdzielczości odpowiadają warstwy konwolucyjne i transponowane, obserwować powinniśmy omawiane artefakty w przestrzeni częstotliwościowej (rys. 8.3.11). Okazuje się jednak, że zastosowanie innej bazy danych uczących (*i.e. Stanford dogs* zamiast FFHQ), skutkuje zanikiem artefaktów w widmie obrazów sztucznych z modelu StyleGAN. Różnica między tymi bazami danych jest taka, że FFHQ zawiera zdjęcia twarzy, a *Stanford dogs* zdjęcia psów.

Wyciągnąć stąd możemy wniosek, że chociaż w generatorze modelu pix2pix za regulację rozdzielczości obrazów odpowiada wartość kroku poszczególnych warstw konwolucyjnych i transponowanych, to uczenie tego modelu na obrazach podpisów powoduje, iż nie występują dyskryminatywne artefakty w widmie jego obrazów sztucznych. Otwartą kwestią pozostaje tutaj, czy przy zastosowaniu innych metod regulacji rozdzielczości w modelach generujących podpisy, omawiana metoda wykrywania sztuczności obrazów również będzie zawodna.



Rysunek 8.3.10. Przykład widma częstotliwościowego obliczonego dla obrazów sztucznych z modelu StyleGAN przy zastosowaniu bilinearnej (dwuliniowej) i binomialnej (dwumianowej) metody podnoszenia rozdzielczości. Modele uczone były na bazie danych FFHQ.

Źródło: J. Frank, T. Eisenhofer, L. Schönherr, A. Fischer, D. Kolossa, T. Holz, *Leveraging Frequency Analysis for Deep Fake Image Recognition*, arXiv, 26 czerwca 2020 r., http://arxiv.org/abs/2003.08685



Rysunek 8.3.11. Przykład widma częstotliwościowego obliczonego dla obrazów sztucznych z modelu StyleGAN przy zastosowaniu bilinearnej (dwuliniowej) metody podnoszenia rozdzielczości, oraz modelu SN-DCGAN, w którym rozdzielczość regulowana jest przez warstwy konwolucyjne i dekonwolucyjne. Modele uczone były na bazie danych *Stanford dogs*.

Źródło: J. Frank, T. Eisenhofer, L. Schönherr, A. Fischer, D. Kolossa, T. Holz, *Leveraging Frequency Analysis for Deep Fake Image Recognition*, arXiv, 26 czerwca 2020 r., http://arxiv.org/abs/2003.08685

8.4. Wnioski. Opracowany model pozwolił na generowanie sztucznych obrazów podpisów. Zapobieżono niskiej zmienności *intra* osobowej oraz zwiększono zakres zmienności *inter* osobowej generowanych podpisów, poprzez warunkowanie modelu na konturowych obrazach podpisów, które stanowiły jego dane wejściowe. Wykazano, że model zdolny jest generować nowe, sztuczne podpisy danej osoby, na podstawie zmodyfikowanych obrazów konturowych pochodzących z oryginalnych podpisów tej osoby. Jakość takich fałszerstw zależna jest jednak nie tylko od skuteczności modelu, ale też od wiadomości pismoznawczych fałszerza.

Poprzez wykorzystanie reprezentatywnej liczby podpisów pochodzących od niewielkiej liczby osób, znacząco obniżono koszty trenowania modelu, jednocześnie uprawdopodobniając możliwość generowania sztucznych podpisów w celu ich fałszowania. Przypuszczać stąd można, że obniżenie liczby probantów do jednego, przy zachowaniu ogólnej liczby podpisów oryginalnych, powinno zapewnić porównywalne rezultaty uczenia modelu. Jednakże, autor przypuszcza, że optymalnym rozwiązaniem byłoby pozyskanie mniejszej liczby podpisów od większej liczby probantów, co zapewniać powinno obniżone koszty uczenia modelu przy zachowaniu jego

skuteczności, a zarazem odpowiadać scenariuszowi, w którym fałszerz dysponuje niewielką liczbą podpisów oryginalnych pochodzących od osoby, która jest jego celem.

Ustalono jednak, że uzyskane podpisy sztuczne nie są wystarczającej jakości, aby spodziewać się wykorzystania podobnej metody w celach fałszerskich. Przede wszystkim, porównując obrazy podpisów sztucznych do prawdziwych wskazać tutaj można: i) różnice w ilości i jakości szumu, oraz innych artefaktów, które mogą przypuszczalnie posłużyć do wykluczenia, że podpis został zeskanowany (wymagałoby to szerszych badań statystycznych na temat szumu i innych artefaktów powodowanych przez wykorzystanie skanerów, aby ustalić czy zaobserwowane różnice ich ilości i jakości stanowią cechy dyskryminatywne); ii) różnice w jakości i ilości cieniowania, które powinny dyskryminować sztuczny obraz podpisu, jeżeli dysponujemy podpisami porównawczymi; iii) różnice ilościowe i jakościowe linii, które dyskryminować powinny obraz sztuczny z punktu widzenia obrazu porównawczego, ale które mogłyby też stanowić samoistne cechy dyskryminatywne (umożliwiać wykluczenie autentyczności obrazu bez potrzeby odwoływania się do materiału porównawczego); iv) zlewanie się blisko położonych linii równoległych lub prostopadłych, oraz pętlic o niewielkiej średnicy, które stanowić powinno samodzielną cechę dyskryminatywną.

Przypuszczać można, że – i) zastosowanie bardziej zaawansowanych architektur sieci generatywno-adwersaryjnych; ii) przy zapewnieniu znacznie większych ilości danych uczących; iii) pochodzących od znacznie większej liczby probantów; iv) przy zapewnieniu wielokrotnie dłuższego czasu nauki modelu – umożliwić powinno wyeliminowanie powyżej wymienionych cech dyskryminatywnych. Autor jest jednak zdania, że koszty takiego przedsięwzięcia byłyby wyższe, niż przypuszczalne korzyści z pozyskanych podpisów sztucznych, którymi operować można tylko w postaci skanów, wydruków i kserokopii, a których przydatność jest ograniczona[314]. Nie można wykluczyć istnienia scenariuszy, w których fałszerstwa takie byłyby opłacalne (*e.g.* celem dezinformacji), ale wydają się one dalece nieprawdopodobne, a metoda w tym kontekście niepraktyczna.

Interesującym problemem jest możliwość wykorzystywania sieci generatywno-adwersaryjnych do generowania instrukcji, na podstawie których inne maszyny (*e.g.* ramiona robotyczne, drukarki 3D, frezarki) mogłyby takie podpisy wykonywać w

314 T. Tomaszewski, *Jeszcze o tym, czy warto badać kopie i inne wtórne odwzorowania pisma ręcznego*, „Człowiek i Dokumenty” nr 42 (2016).

świecie fizycznym. Jeżeli wykluczyć, że maszyna wykonująca podpis powodować będzie cechy dyskryminatywne, to problem wykrywania takich fałszerstw polegał będzie na niedoskonałościach związanych z generowaniem instrukcji przez sieci generatywno-adwersaryjne. Uczenie takich sieci będzie niezwykle trudne, kiedy danymi uczącymi będą podpisy statyczne, ponieważ sieć potrzebowała będzie dostępu do bieżących, fizycznych rezultatów swojego uczenia (sukcesywnie digitalizowanych), a to bardzo istotnie wydłuży proces uczenia. Uczenie takich sieci będzie relatywnie łatwe, kiedy danymi uczącymi będą podpisy dynamiczne, ponieważ wystarczy, że sieć generować będzie fałszywe podpisy dynamiczne, skoro te same w sobie stanowić mogą instrukcje do wykonywania podpisów w świecie fizycznym. Zakładając jednak, że pomiędzy podpisami statycznymi i dynamicznymi danej osoby występują cechy dyskryminatywne, to fizyczne fałszerstwa podpisów statycznych, przeprowadzane z wykorzystaniem sieci generatywno-adwersaryjnych uczonych na podpisach dynamicznych, będą wykrywalne za pomocą badań pismoznawczych.

8.5. Reprodukowalność. Wszystkie szczegóły, metody, rezultaty, dane i dodatkowe objaśnienia zostały udokumentowane i dostępne są w repozytorium autora na platformie Github: https://github.com/Ma-Marcinowski/Generative-Model

Modele uczono i testowano za pomocą chmury obliczeniowej Google Colab(oratory), umożliwiającej korzystanie z GPU Nvidia Tesla K80. Gdzie przeciętna epoka uczenia prezentowanego wariantu modelu trwała 260 sekund (prezentowany wariant modelu trenowano 260 epok).

Rozdział 9. Dyskusja.

9.1. Przyszłość sztucznych sieci neuronowych w kryminalistyce. Autor zebrał i uporządkował wnioski uzasadnione wynikami przeprowadzonych badań, prezentując je w kontekście pozytywnych, neutralnych i negatywnych przesłanek dla zastosowań sztucznych sieci neuronowych w kryminalistyce[315]. Gdzie: i) przesłanki pozytywne stanowią o generalnych zaletach uczenia maszynowego w kryminalistyce; ii) przesłanki neutralne traktują o takich okolicznościach, które nie są ani zaletami, ani wadami sieci neuronowych, ale ograniczają ich zastosowania, bo choć można im zaradzić, to jest to trudne i kosztowne; iii) przesłanki negatywne uniemożliwiać lub zniechęcać będą wobec zastosowania sieci neuronowych do badań kryminalistycznych.

Przesłanki pozytywne.

***Uniwersalność*.** Sztuczne sieci neuronowe, które spełniają określone postulaty (*i.a.* są wielowarstwowe, jednokierunkowe, nieliniowe, *etc*.), uznawane są za uniwersalne aproksymatory funkcji (*i.e*. mogą przybliżyć się do dowolnej funkcji odwzorowującej dowolny zbiór na dowolny inny zbiór)[316]. Dokonywać tego mogą na podstawie: i) danych surowych; ii) danych automatycznie preprocesowanych; iii) danych predeterminowanych przez ekspertów; iv) danych pochodzących od innych modeli uczenia maszynowego; v) danych stanowiących kombinacje punktów i-iv. Ogólnie rzecz biorąc, sztuczne sieci neuronowe stanowią uniwersalne i automatyczne narzędzia rozwiązywania problemów, możliwe do zastosowania w dowolnej dziedzinie, pod warunkiem dostarczenia danych o odpowiedniej jakości i ilości.

***Obiektywizm*.** Potencjalnie, sztuczne sieci neuronowe służyć mogą do automatycznego rozwiązywania problemów w sposób obiektywny, ale stwierdzenie obiektywności wymaga, aby model był interpretowalny i ewaluowalny. Obiektywizm, którego powinniśmy oczekiwać od modelu kryminalistycznego, powinien być natury empirycznej i naukowej[317]. Nawet jeżeli model uczono na miarodajnej ilości danych

315 Tematyka ta została też częściowo przedstawiona w artykule: M. Marcinowski, *Perspektywy wykorzystania sztucznych sieci neuronowych w badaniach kryminalistycznych* [w:] *Inspiracje i wspomnienia dedykowane pamięci Jerzego Koniecznego*, Kraków 2021, s. 123–145.

316 B.C. Csáji, *Approximation with Artificial Neural Networks*, Uniwersytet Eötvös Loránd 2001.

317 M. Goc, A. Łuszczuk, K. Łuszczuk, T. Tomaszewski, *Programy komputerowe jako narzędzie wspomagające ekspertyzę pisma ręcznego*, „Problemy Kryminalistyki.” (2016).

przetworzonych z udziałem ekspertów, którzy użyczyli mu w ten sposób swojej wiedzy i doświadczenia, to model taki będzie co najwyżej intersubiektywny, a nie obiektywny. Aby zapewnić, że model jest obiektywny, *i.e.* że realizuje metody naukowe i realizuje je poprawnie, model taki musi być interpretowalny i ewaluowalny.

Powszechnie określa się modele uczenia maszynowego jako obiektywne, mając na myśli, że są one niezależne od decyzji i wpływów człowieka. Niestety, decyzje modelu często wymagają interpretacji i nadzoru. Ponadto, istotny wpływ na decyzje modelu wywierać można manipulując ilością i jakością danych wejściowych. Stosować można też zaawansowane metody, *e.g.* zatruwanie danych, ataki adwersaryjne lub tylne drzwi (*data poisoning, adversarial attacks, backdoors*).

***Zupełność*.** Potencjalnie uniwersalne i obiektywne, sztuczne sieci neuronowe mogą być zupełnymi i zamkniętymi systemami rozwiązywania problemów (tzw. metody od-początku-do-końca, *end-to-end solutions*). Celem rozwiązania problemu sieć neuronowa nie wymaga pomocy lub ingerencji człowieka, a jedynie tego, aby człowiek wprowadził do niej surowe dane wejściowe. W przypadku danych wymagających wcześniejszego przetworzenia, będzie potrzebna pomoc ekspertów. Przy czym, jeżeli istnieją już specjalistyczne bazy danych wejściowych i wyjściowych (*e.g.* zdjęcia fMRI i diagnozy ekspertów), to nawet na etapie tworzenia modelu udział ekspertów nie jest konieczny, choć jest bardzo wskazany dla ewaluacji maszyny.

Przesłanki neutralne.

***Dostępność danych*.** Podstawowym problemem z pozyskiwaniem danych do uczenia sztucznych sieci neuronowych jest ich koszt, przy czym dane surowe lub automatycznie preprocesowane są wielokrotnie mniej kosztowne, niż dane przetwarzane i oznaczane przez ekspertów.

Inny problem stanowić muszą dane osobowe, które są najczęściej danymi wrażliwymi (*e.g.* dane biometryczne i medyczne), podlegają więc szczególnym rygorom (*e.g.* Ogólne rozporządzenie o ochronie danych) i reglamentowane są zainteresowanym badaczom po spełnieniu określonych warunków (*e.g. UK Biobank*)[318].

318 C. Bycroft, C. Freeman, D. Petkova, G. Band, L.T. Elliott, K. Sharp, A. Motyer, D. Vukcevic, O. Delaneau, J. O'Connell, A. Cortes, S. Welsh, A. Young, M. Effingham, G. McVean, S. Leslie, N. Allen, P. Donnelly, J. Marchini, *The UK Biobank resource with deep phenotyping and genomic data*, „Nature" t. 562 nr 7726 (2018), DOI: 10.1038/s41586-018-0579-z.

Kolejnym wyzwaniem jest reprezentatywność statystyczna danych uczących, która sama w sobie jest czasochłonna i trudna do osiągnięcia, a spotkać się można z sytuacjami intencjonalnego manipulowania danymi przez zbierających (w tym organy publiczne, tzw. *dirty data*)[319]. Jeżeli manipulacja danymi zostanie nawet wykryta, to na ogół nie będzie można ich już wykorzystać.

Ostatecznie, nawet jeżeli pozyskano obiektywne i statystycznie reprezentatywne dane, to mogą one być etycznie i prawnie niedopuszczalne do zastosowania, będąc reprezentatywne wobec niesprawiedliwej społecznie rzeczywistości (*e.g.* z powodu strukturalnego rasizmu)[320]. Dlatego, że model uczenia maszynowego stworzony na podstawie danych reprezentatywnych wobec rzeczywistości dyskryminatywnej, będzie powielał, wzmacniał i konserwował tą rzeczywistość, często przecież uchybiającą prawom człowieka.

***Miarodajność danych*.** Nawet jeżeli uda się pozyskać odpowiednie dane, to reprezentatywność statystyczna jest niewystarczające wobec modeli uczenia maszynowego. Dlatego, że przypadki, które model uczenia maszynowego miałby rozstrzygać, różnić się będą w poziomach trudności, klasach rozwiązań i przynależności do subpopulacji. Przypadki te mogą być dysproporcjonalnie reprezentowane na zbiorach danych uczących (zarówno pod względem ilości i jakości danych), nawet gdyby zbiory te były statystycznie reprezentatywne. Podobny pogląd reprezentowany jest w projekcie Aktu w sprawie sztucznej inteligencji, gdzie art. 10 i 13 wymagają, aby uwzględniać różnice pomiędzy poszczególnymi grupami ludzi i określać indywidualne poziomy trafności dla tych poszczególnych grup osób.

Na przykład, gdyby nauczyć model weryfikacji pisma ręcznego na próbie statystycznie reprezentatywnej, to model taki nauczy się nade wszystko weryfikacji pisma osób praworęcznych, bo w przeciwieństwie do pisma osób leworęcznych, będzie ono miało największy wpływ na jego trafność. Model taki będzie uczył się poprawnej weryfikacji pisma osób leworęcznych w zakresie tożsamym z weryfikacją pisma osób leworęcznych, oraz w dalszej kolejności.

319 R. Richardson, J.M. Schultz, K. Crawford, *Dirty Data, Bad Predictions: How Civil Rights Violations Impact Police Data, Predictive Policing Systems, and Justice*, „New York University Law Review" t. 94 (2020).

320 L. Bennett-Moses, J. Chan, *Algorithmic prediction in policing: assumptions, evaluation, and accountability*, „Policing and Society" t. 28 (2018).

Istnieje stosunkowo niewiele sposobów pozwalających ten problem umniejszyć i zapewnić miarodajność danych. Jeżeli obserwuje się dysproporcjonalny rozkład ilości i jakości danych dostępnych dla: i) różnych poziomów trudności przypadków; ii) różnych klas rozwiązań przypadków; iii) różnych subpopulacji do których przypadki należą. To można *i.a.*: i) tworzyć dla odrębnych problemów odrębne modele uczenia maszynowego; ii) niwelować dysproporcje poprzez umniejszaną lub powiększaną reprezentację danego problemu na zbiorze uczącym; iii) niwelować dysproporcje za pomocą funkcji kosztu (*e.g.* zwiększając koszt błędów popełnianych względem mniej częstych lub trudniejszych przypadków). Pierwsze rozwiązanie byłoby z jednej strony kosztowne, a z drugiej budziłoby kolejne wątpliwości (*e.g.* czy nie jest dyskryminatywnym zastosowanie innego modelu wobec większości, a innego wobec mniejszości). Drugie rozwiązanie powodowałby konieczność zaniechania dużych ilości lub jakości danych, poprzez ograniczenie się do ilości i jakości dostępnych dla niedoreprezentowanych przypadków. Trzecie rozwiązanie nie niesie ze sobą oczywistych wad, ale może być tylko częściowo skuteczne.

Przesłanki negatywne.

***Nieinterpretowalność*.** Procesy decyzyjne sztucznych sieci neuronowych są nieinterpretowalne (*uninterpretable*), co jest też czasem określane jako niewymienialność lub niewytłumaczalność (*unexplainable* lub *unexplicable*), dlatego też są obrazowo nazywane czarnymi skrzynkami (*black-boxes*)[321]. Przyczyna polega na tym, że owe procesy decyzyjne – relacje zachodzące pomiędzy neuronami sieci – są zbyt liczne i skomplikowane, aby człowiek zdolny był nadać im znaczenie semantyczne (*i.e.* aby zdolny był je nazwać).

Nieinterpretowalne są również procesy neuronalne zachodzące w mózgu człowieka (na poziomie podświadomości), które to dopiero determinują jego semantycznie sensowne myśli i decyzje (zachodzące na poziomie świadomości). Rezultaty ludzkich procesów podświadomych są na poziomie świadomości nazywalne, i tymi nazwami człowiek operuje – za pomocą procesów podświadomych – konstruując wyjaśnialne dla siebie decyzje i ich przyczyny. Podczas gdy, rezultaty procesów

321 K. Atkinson, T. Bench-Capon, D. Bollegala, *Explanation in AI and law: Past, present and future*, „Artificial Intelligence" t. 289 (2020).

neuronalnych zachodzących w sieciach neuronowych są nazwami (dane wyjściowe), na podstawie których nie da się konstruować ich wyjaśnień.

Problem interpretowalności sztucznych sieci neuronowych nie został dotąd rozwiązany w sposób uniwersalny. Najbardziej popularne rozwiązania polegają na obliczaniu wstecznym, który to fragment danych miał decydujący wpływ na dane rozstrzygnięcie. Jest to jednak metoda zawodna[322], ponieważ, nawet kiedy wiemy co ten fragment danych oznacza dla ludzi, to nie wiemy co on oznacza dla maszyny.

Istnieją natomiast obiecujące rozwiązania szczególne, które związane są na przykład z zastosowaniem sztucznych sieci neuronowych w prawie[323]. Przykładowo, jeżeli na podstawie orzeczeń można nauczyć model formułować orzeczenia, to i na podstawie opinii biegłych można nauczyć model formułować opinie. Ściślej rzecz biorąc, gdyby model miał weryfikować pismo ręczne i uzasadniać swoje decyzje, to rozwiązać należy dwa problemy: i) nauczyć model weryfikacji podpisów na przykładach podpisów; ii) równocześnie nauczyć model uzasadniania swoich rozstrzygnięć weryfikacyjnych na przykładzie opinii biegłych. Nie będzie jednak gwarancji, że nawet najlepiej skonstruowane przez model uzasadnienia będą tożsame z tymi procesami decyzyjnymi, które determinują jego rozstrzygnięcia weryfikacyjne. Innymi słowy, będą one raczej tylko skorelowane, niż związane przyczynowo-skutkowo (współ-determinujące-się). Uogólniając, będzie to interpretowalność powierzchowna (*i.e.* na poziomie semantycznie sensowych danych wyjściowych, które być może wyjaśniają procesy neuronalne będące ich przyczyną). Nie będzie to jednak interpretowalność głęboka (*i.e.* na poziomie procesów neuronalnych). Interpretowalność powierzchowną porównać można do wyjaśnień, którymi człowiek dysponuje na poziomie świadomym, a interpretowalność głęboką do niedostępnych człowiekowi wyjaśnień procesów zachodzących podświadomie.

Istota problemu polega na tym, że nawet jeżeli model został poddany rzetelnej ewaluacji i dowiedziono ponad rozsądną wątpliwość, że jest trafny, to nadal nie są znane przyczyny dla decyzji podejmowanych przez model. Stąd: i) nie wiadomo czy decyzje te wynikają z powodów naukowych i racjonalnych; ii) nie wiadomo w jakich

322 C. Rudin, *Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead*, „Nature Machine Intelligence" t. 1 nr 5 (2019), DOI: 10.1038/s42256-019-0048-x.

323 L.K. Branting, C. Pfeifer, B. Brown, L. Ferro, J. Aberdeen, B. Weiss, M. Pfaff, B. Liao, *Scalable and explainable legal prediction*, „Artificial Intelligence and Law" t. 29 nr 2 (2021), DOI: 10.1007/s10506-020-09273-1.

przypadkach i sytuacjach metody te będą zawodne, a model nie powinien być stosowany; iii) trudno jest podważać to, czego nie potrafi się nazwać. Jak wykazano na przykładzie sztucznej sieci neuronowej, którą opisano w rozdziale 7, interpretowalność powierzchowna udzielić może odpowiedzi na te niewiadome. Biorąc pod uwagę projekt Aktu w sprawie sztucznej inteligencji, zapewnienie przynajmniej powierzchownej interpretowalności sztucznych sieci neuronowych będzie konieczne w perspektywie zastosowań kryminalistycznych, skoro art. 13 nakładać będzie obowiązek przejrzystości działania modelu, która umożliwiać miałaby interpretację wyników jego działania i właściwe ich wykorzystanie.

***Niefalsyfikowalność*.** Problem nieinterpretowalności sztucznych sieci neuronowych pociąga za sobą ograniczoną ewaluowalność tych maszyn, na którą składają się weryfikacja (sprawdzanie pozytywne) i falsyfikacja (sprawdzanie negatywne). Ograniczenie ewaluowalności sieci neuronowych jest jednostronne, ponieważ ulegają one bardzo łatwej weryfikacji (potwierdzaniu) poprzez ogólne testowanie i raportowanie wskaźników trafności, a z drugiej strony bardzo trudno poddać je falsyfikacji (obalaniu).

Procesy decyzyjne sztucznych sieci neuronowych są nieinterpretowalne, bo nie posiadają znaczenia semantycznego, trudno więc z nich wyprowadzać hipotezy falsyfikujące (zdania), a przynajmniej wyprowadzać je w satysfakcjonującej ilości i jakości, przez co są niemalże niefalsyfikowalne. Jeżeli więc zwrócić uwagę na łatwą weryfikowalność tych modeli, to jedyną możliwością na udowodnienie prawidłowości modelu jest próba jego falsyfikacji. Gdy falsyfikacja się powiedzie, można taki model odrzucić. Kiedy się zaś nie powiedzie, pomimo najlepszych usiłowań, to uznać można, iż model tymczasowo udowodniono jako dopuszczalną metodę badań kryminalistycznych. Stąd w rozdziale 6 przeprowadzono przykłady obalania sztucznych sieci neuronowych poprzez dowodzenie ich nierzetelności na różnych grupach przypadków, różnych poziomach trudności, oraz wobec różnych subpopulacji. Zważając na wymogi odwołujące się do trafności i rzetelności, które stawiał będzie Akt w sprawie sztucznej inteligencji w artykułach 10, 13 i 15, najbardziej przydatnym sposobem ewaluacji sztucznych sieci neuronowych będzie próba ich falsyfikacji.

***Niestabilność*.** Jedną z największych słabości sztucznych sieci neuronowych jest ich podatność na perturbacje i zanieczyszczenia danych, które spowodować mogą: i) losowe rozstrzygnięcia modelu; ii) lub rozstrzygnięcia na rzecz określonej klasy. Ponadto, na etapie uczenia maszyny obecność zaszumionych danych spowodować może, iż model nie nauczy się rozstrzygać na podstawie racjonalnych względów i metod, ale zanieczyszczeń skorelowanych z rozwiązaniami.

Jednym z objawów niestabilności sztucznych sieci neuronowych jest ich podatność na fałszerstwo. Ponieważ, wywierać można wpływ na decyzje modelu poprzez: i) proste manipulacje ilością i jakością danych wejściowych (*e.g.* nakładanie losowego szumu); ii) ataki adwersaryjne (*adversarial attacs*) polegające na obliczeniu wstecznym takiego szumu, którego naniesienie na dane wejściowe pozwoli uzyskać określone rozstrzygnięcie.

Dlatego też trudno byłoby zaakceptować sztuczne sieci neuronowe w praktyce kryminalistycznej, bez uprzedniego określenia metodyki ich zastosowań. Na przykładzie badań przeprowadzonych w rozdziale 6, wykazano, że zastosowanie sztucznej sieci neuronowej do przetwarzania danych, które pochodzą z innych źródeł niż jej dane uczące, poprzedzone być powinno ewaluacją modelu na podstawie tych danych (w innym przypadku nie będzie znana rzeczywistej trafność maszyny wobec tych danych). Wnioski te są tożsame z wymogami stawianymi przez projekt Aktu w sprawie sztucznej inteligencji, którego art. 10, 13 i 15 wymagają zapewnienia odpowiedniego poziomu cyberbezpieczeństwa i rzetelności modelu, oraz określenia warunków wpływających na te poziomy w instrukcji obsługi, podług której zobowiązany będzie postępować użytkownik.

***Niepowtarzalność*.** Problem reprodukowalności i replikowalności sieci neuronowych jest specyficzny i trudny do uniknięcia. Ale w mniejszym stopniu dotyczył on będzie zastosowań sztucznych sieci neuronowych w kryminalistyce, a w większym stopniu badań zmierzających do dopuszczenia ich do takich zastosowań.

Jeżeli rozumieć reprodukowalność badań, jako uzyskanie tego samego wyniku na podstawie tych samych danych i tymi samymi metodami[324], to sieci neuronowe

324 M. Miłkowski, W.M. Hensel, M. Hohoł, *Replicability or reproducibility? On the replication crisis in computational neuroscience and sharing only relevant detail*, „Journal of Computational Neuroscience" t. 45 (2018).

postulatu reprodukowalności nie spełniają. Przede wszystkim, już parametry modelu, które podlegają optymalizacji w skutek jego nauki, są początkowo losowe. Ponadto, dla uniknięcia przeuczenia modelu, *e.g.* zapamiętania przez model kolejności prawidłowych odpowiedzi, dane uczące są losowo tasowane pomiędzy epokami uczenia modelu. Z przyczyn tych, zastosowanie takich samych danych i metod nie gwarantuje uzyskania tego samego modelu. Problem ten pogłębia jeszcze nieinterpretowalność i słaba falsyfikowalność sieci neuronowych, sprawiając, że skutkiem reprodukcji będą słabo porównywalne modele.

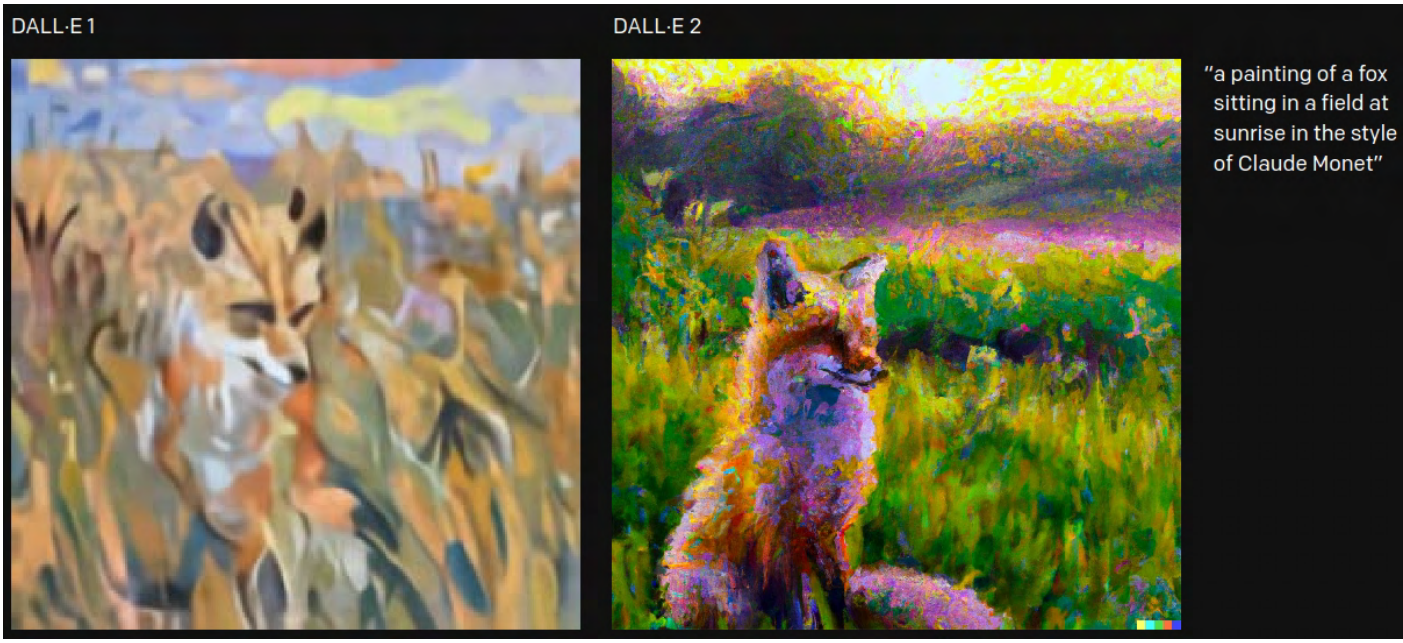
Jeżeli rozumieć replikowalność badań, jako powtarzalność wyników na podstawie nowych danych[325], to sieci neuronowe nie spełniają tego postulatu, skoro zastosowanie nowych danych uczących z pewnością spowoduje odmienne i słabo porównywalne wobec siebie modele sieci neuronowych.

Nie są te problemy nierozwiązywalne i można im zaradzić nakładając obowiązek, *e.g.*: i) aby stosować takie modele, które przeszły zewnętrzną i niezależną od swoich twórców falsyfikację, reprodukcję i replikację; ii) aby stosujący dany model poddawał go ewaluacji w warunkach swojego warsztatu i danego przypadku, do którego model będzie stosował; iii) aby nie ograniczać się do zastosowania jednego modelu, ale stosować ich jak największą ilość i jakość, zmierzając przy tym do osiągnięcia wyników sprzecznych, które podważałyby decyzje tych maszyn, a więc i słuszność ich zastosowania. Postulaty te będą częściowo zbieżne z obowiązkami, które nakładał będzie art. 29 projekt Aktu w sprawie sztucznej inteligencji, wymagając aby użytkownik modelu: i) zapewnił adekwatność danych wejściowych; ii) monitorował działanie modelu; iii) sprawował nadzór nad modelem; iv) postępował według instrukcji obsługi, określającej okoliczności, które mogą mieć wpływ na oczekiwany poziom trafności, rzetelności i cyberbezpieczeństwa modelu.

9.2. Przyszłość sztucznych sieci neuronowych w antykryminalistyce. Niektórzy autorzy są zdania, iż – i) wzrost ilości i jakości danych sztucznie generowanych; ii) oraz ułatwienie ich wytwarzania i utrudnienie ich wykrywania – spowoduje wzrost zaufania publicznego, ponieważ zmusi społeczeństwo do aktywnego wyszukiwania godnych zaufania źródeł (*i.e.* zwiększenie zaufania wynikać będzie ze

325 Ibid.

zmniejszenia wiedzy)[326]. Większość badaczy uważa jednak, że sytuacja upadku zaufania publicznego do danych cyfrowych byłaby wysoce niebezpieczna, podobnie jak dezinformacja wynikająca z ufności publicznej w fałszywe informacje cyfrowe[327]. Różnica polega tutaj na odmiennych definicjach zaufania i przewidywanych skutkach jego upadku lub wzrostu. Wydaje się jednak, że obydwa poglądy są zgodne co do tego, że utrata wiedzy o prawdziwości danych, spowodowana upowszechnieniem się fałszerstw głębokich, spowoduje konieczność ufania w wybrane sobie źródła danych i podmioty oceniające ich prawdziwość.



Rysunek 9.2.1. Porównanie przykładowych obrazów wygenerowanych przez sztuczne sieci neuronowe DALL·E i DALL·E 2. Po prawej widoczny jest tekst wejściowy, na podstawie którego sieci te wygenerowały widoczne obrazy.

Źródło: https://openai.com/dall-e-2/, dostęp 12 stycznia 2023.

326 H. Etienne, *The future of online trust (and why Deepfake is advancing it)*, „AI and Ethics" t. 1 nr 4 (2021), DOI: 10.1007/s43681-021-00072-1.

327 S. Agarwal, H. Farid, Y. Gu, M. He, K. Nagano, H. Li, *Protecting World Leaders Against Deep Fakes* [w:] *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2019; M. Pawelec, *Deepfakes and Democracy (Theory): How Synthetic Audio-Visual Media for Disinformation and Hate Speech Threaten Core Democratic Functions*, „Digital Society" t. 1 nr 2 (2022), DOI: 10.1007/s44206-022-00010-6; M. Tomaszewska-Michalak, *Fake news – wstępna analiza zjawiska*, „Przegląd Politologiczny" nr 1 (2021), DOI: 10.14746/pp.2021.26.1.4; C. Vaccari, A. Chadwick, *Deepfakes and Disinformation: Exploring the Impact of Synthetic Political Video on Deception, Uncertainty, and Trust in News*, „Social Media + Society" t. 6 nr 1 (2020), DOI: 10.1177/2056305120903408.

Obecne trendy wskazują na to, że dane wytwarzane za pomocą sztucznych sieci neuronowych, będą coraz wyższej jakości i trudniejsze do wykrycia[328]. Dobrym przykładem powyższych przypuszczeń są sieci DALL·E[329] i DALL·E 2[330]. Pierwszą z nich opublikowano w lutym 2021 roku, a drugą w kwietniu 2022 roku. Gdzie, model późniejszy, opracowany w relatywnie niewielkim odstępie czasu od modelu wcześniejszego, nauczono generować obrazy o znacząco wyższej jakości (rys. 9.2.1).

W perspektywie kryminalistycznej, problem wysokiej jakości i trudności wykrywania fałszerstw głębokich może okazać się drugorzędny. Jeżeli dane sztuczne będą: i) na tyle wysokiej jakości, aby inspekcja wzrokowa nie umożliwiała ich dyskryminacji; ii) oraz wykrycie ich będzie możliwe tylko przy zaangażowaniu eksperta; iii) a metody ich generowania będą przy tym łatwo dostępne i niekosztowne. To, istotą problemu będzie potencjalnie wielka liczba, a nie jakość tych fałszerstw.

Przypuszczać można, że będą to w przeważającej większości fałszerstwa głębokie o charakterze pornograficznym[331], w tym jednak przypadku ustalenie ich autentyczności będzie drugorzędne względem konieczności ochrony praw osób, których wizerunek został wykorzystany, oraz wykrycia i ukarania osób rozpowszechniających takie materiały. Przy tym, pociągnięcie do odpowiedzialności za rozpowszechnianie nagiego wizerunku osoby bez jej zgody, może okazać się niemożliwe, jeżeli autentyczność danych zostanie skutecznie zakwestionowana (*i.e.* w przypadkach, gdy wizerunek osoby pokrzywdzonej naniesiony została na nagie ciało innej osoby)[332].

Uogólniając powyższą obserwację, od kwalifikacji prawnej zależeć będzie, czy autentyczność danych będzie kwestionowana. Na przykład, w przypadku naruszeń praw autorskich i danych osobowych, spowodowanych rozpowszechnianiem fałszerstw głębokich, kwestia ich autentyczności będzie drugorzędna. Podczas gdy, w przypadkach znieważenia, zniesławienia, fałszerstwa dokumentu, kradzieży tożsamości lub naruszenia dóbr osobistych, kwestia oceny autentyczności danych sztucznych będzie na

328 Y. Mirsky, W. Lee, *The Creation and Detection of Deepfakes: A Survey*, 31 stycznia 2022 r., http://arxiv.org/abs/2004.11138.

329 A. Ramesh, M. Pavlov, G. Goh, S. Gray, C. Voss, A. Radford, M. Chen, I. Sutskever, *Zero-Shot Text-to-Image Generation*, arXiv, 26 lutego 2021 r., http://arxiv.org/abs/2102.12092.

330 A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, M. Chen, *Hierarchical Text-Conditional Image Generation with CLIP Latents*, arXiv, 12 kwietnia 2022 r., http://arxiv.org/abs/2204.06125.

331 H. Etienne, *The future of online trust (and why Deepfake is advancing it)*, „AI and Ethics" t. 1 nr 4 (2021), DOI: 10.1007/s43681-021-00072-1, s. 7.

332 A. Ziobroń, *Deepfake a prawo karne. Uwagi „de lege lata" i „de lege ferenda" dotyczące fałszywej pornografii*, „Studenckie Prace Prawnicze, Administratywistyczne i Ekonomiczne" t. 37 (2021), DOI: 10.19195/1733-5779.37.15, s. 229–230.

ogół pierwszorzędna. Niekiedy zaś, kwalifikacja prawna zależeć będzie od tego, czy autentyczność danych będzie skutecznie kwestionowana.

Dalej, dokonać można podziału na dane sztuczne, które: i) stanowią dowód popełnienia czynu (*e.g*. fałszywe nagranie z miejsca zdarzenia); ii) stanowią narzędzie za pomocą którego czyn został popełniony (*e.g.* bezprawna groźba rozpowszechnienia fałszywych materiałów wideo). W pierwszym przypadku ocena autentyczności danych będzie konieczna, w drugim zaś będzie zależna od kwalifikacji prawnej czynu.

Uwzględniając powyższe obserwacje, przypuszczać można, że wzrost liczby fałszerstw głębokich (zakładając wzrost ich jakości, łatwości opracowania i trudności wykrycia), spowoduje raczej zwiększenie liczby postępowań, niż liczby wymaganych dlań ekspertyz (badających autentyczność danych cyfrowych). Przy tym, wzrost jakości i trudności wykrywania fałszerstw głębokich spowodować może wydłużenie czasu koniecznego do wydania opinii.

Otwartym problemem pozostaje tutaj zagadnienie automatycznego wykrywania fałszerstw głębokich. Z jednej strony gwarantuje ono znacznie ułatwienie i przyśpieszenie ekspertyz, z drugiej zaś takie sieci neuronowe obarczone są nieinterpretowalnością i niską ewaluowalnością. Ponadto, modele takie posłużyć mogą jako dyskryminatory w modelach fałszujących, zapewniając im generowanie jeszcze lepszych fałszerstw. Aby temu zapobiec, można unikać ich publikacji i udostępniania, ale czyniąc to kosztem transparentności. Ponadto, modele takie podatne są na ataki adwersaryjne, które mogą być podobnie efektywne, niezależnie od tego czy posiada się dostęp do atakowanego modelu, czy też nie.

Podsumowanie

Aspekty praktyczne i metodologiczne związane z zastosowaniami sztucznych sieci neuronowych w kryminalistyce autor przedstawił pod postacią trzech pytań badawczych:

I) Jak dokonywać ewaluacji sztucznych sieci neuronowych dla potrzeb kryminalistyki?

II) Jak dokonywać interpretacji sztucznych sieci neuronowych dla potrzeb kryminalistyki?

III) Jak wykrywać fałszerstwa popełniane z wykorzystaniem sztucznych sieci neuronowych?

Autor udzielił odpowiedzi na postawione pytania, na przykładzie własnych badań empirycznych, rozważając ich wyniki w kontekście prawnym, praktycznym i metodologicznym.

Odpowiadając na pierwsze pytanie, autor przeprowadził badania empiryczne, gdzie przykład badań stanowiły sztuczne sieci neuronowe do weryfikacji wykonawstwa dokumentów odręcznie pisanych. W oparciu o wyniki tych badań, autor wskazał, że najbardziej skuteczną metodą ewaluacji sztucznych sieci neuronowych jest poddanie ich próbie falsyfikacji. Ponieważ, pozwala ona podważyć rzetelność modelu, poprzez wykazanie, że w różnych okolicznościach osiąga on istotnie różne poziomy trafności. Okoliczności te autor ustalał w dwójnasób: i) wyznaczając arbitralne lub obiektywne podzbiory na zbiorze testowym; ii) dokonując celowych manipulacji danymi wejściowymi. W pierwszym przypadku, autor wyznaczał: i) kryteria ewaluacji (arbitralne), gdzie testował modele na arbitralnych kombinacjach podzbiorów, dążąc do wykrycia nierzetelności; ii) kategorie ewaluacji (obiektywne), gdzie wyróżniał podzbiory testowe ze względu na przynależność rozstrzyganych przypadków do różnych subpopulacji. W drugim przypadku, autor przeprowadzał ewaluację ze względu na: i) różne ilości danych testowych; ii) różne jakości danych testowych; iii) różną trudność przypadków testowych; iv) różne metody preprocesowania danych wejściowych; v) różne źródła danych uczących i testowych.

Odpowiadając na drugie pytanie, autor przeprowadził badania empiryczne, gdzie przykład stanowiły sztuczne sieci neuronowe do identyfikacji wykonawcy dokument.

Autor opracował w tym celu oryginalną metodę, którą jest powierzchownie interpretowalna sieć neuronowa. Gdzie, interpretację powierzchownego modelu przeprowadza się na podstawie danych wyjściowych, które są hierarchicznie zorganizowane i związane przyczynowo-skutkowo. W tym kontekście autor omówił i zastosował opisywane w literaturze metody interpretacji sztucznych sieci neuronowych, wykazując ich ograniczenia. Jednakże, z ich pomocą autor poddał w wątpliwość rezultaty uczenia modelu powierzchownie interpretowalnego, podkreślając łatwość falsyfikacji, jaką można przeprowadzić wobec takiego modelu.

Odpowiadając na trzecie pytanie, autor przeprowadził badania empiryczne, gdzie przykład stanowiła sieć generatywno-adwersaryjna do fałszowania podpisów. Autor opracował tutaj model hybrydowy, który umożliwia istotny wpływ użytkownika na wynik generowania podpisu przez model. Dzięki temu, niezwykle trudne zadanie, jakim jest sfałszowanie podpisu, autor rozłożył na dwa proste etapy, gdzie: i) człowiek określa kontur nowego podpisu, modyfikując podpisy oryginalne; ii) a model wypełnia kontur odpowiednią teksturą. Autor przeprowadził następnie analizę podpisów sztucznych, wykazując cechy sugerujące ich fałszywość, które określał za pomocą terminologii pismoznawczej. Autor potwierdził w ten sposób, że możliwym jest ujawnianie fałszerstw głębokich za pomocą metod kryminalistycznych, które się w tym nie specjalizują.

Podsumowując, jeżeli sztuczne sieci neuronowe miałyby być stosowane w praktyce kryminalistycznej, to powinny one być ewaluowane poprzez falsyfikację ich rzetelności i przynajmniej powierzchownie interpretowalne. W zakresie wykrywania antykryminalistycznych zastosowań sztucznych sieci neuronowych, obecne metody badań kryminalistycznych powinny być nadal skuteczne do wykrywania fałszerstw danych, których badaniem się zajmują, nawet jeżeli nie specjalizują się w wykrywaniu fałszerstw głębokich.

Bibliografia

1. AbdulNabi I., Yaseen Q., *Spam Email Detection Using Deep Learning Techniques*, „Procedia Computer Science" t. 184 (2021), DOI: 10.1016/j.procs.2021.03.107.
2. Afchar D., Nozick V., Yamagishi J., Echizen I., *MesoNet: a Compact Facial Video Forgery Detection Network* [w:] *2018 IEEE International Workshop on Information Forensics and Security (WIFS)*, 2018.
3. Agarwal S., Farid H., Gu Y., He M., Nagano K., Li H., *Protecting World Leaders Against Deep Fakes* [w:] *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2019.
4. Ahmed R., Dashtipour K., Gogate M., Raza A., Zhang R., Huang K., Hawalah A., Adeel A., Hussain A., *Offline Arabic Handwriting Recognition Using Deep Machine Learning: A Review of Recent Advances* [w:] J. Ren, A. Hussain, H. Zhao, K. Huang, J. Zheng, J. Cai, R. Chen, Y. Xiao (red.), *Advances in Brain Inspired Cognitive Systems*, Cham 2020.
5. Alali Y., Harrou F., Sun Y., *A proficient approach to forecast COVID-19 spread via optimized dynamic machine learning models*, „Scientific Reports" t. 12 nr 1 (2022), DOI: 10.1038/s41598-022-06218-3.
6. Ali M., Sapiezynski P., Bogen M., Korolova A., Mislove A., Rieke A., *Discrimination through optimization: How Facebook's ad delivery can lead to skewed outcomes*, „Proceedings of the ACM on Human-Computer Interaction" t. 3 nr CSCW (2019), DOI: 10.1145/3359301.
7. Ali M., Sapiezynski P., Korolova A., Mislove A., Rieke A., *Ad Delivery Algorithms: The Hidden Arbiters of Political Messaging*, „arXiv:1912.04255 [cs]" (2019), http://arxiv.org/abs/1912.04255.
8. Amidi A., Amidi S., *Deep Learning; Convolutional Neural Networks. Stanford University* [na:] https://stanford.edu/~shervine/teaching/cs-230/cheatsheet-convolutional-neural-networks, 2019 r., dostęp 12 stycznia 2023 r.
9. Atkinson K., Bench-Capon T., Bollegala D., *Explanation in AI and law: Past, present and future*, „Artificial Intelligence" t. 289 (2020).
10. Babcock T., Richmond G., Spooner M., Holmes W., *Application of Perceptrons to Photointerpretation*, Buffalo 1964.

11. Bakator M., Radosav D., *Deep Learning and Medical Diagnosis: A Review of Literature*, „Multimodal Technologies and Interaction” t. 2 nr 3 (2018), DOI: 10.3390/mti2030047.
12. Baldi P., Sadowski P.J., *Understanding Dropout* [w:] *Advances in Neural Information Processing Systems*, t. 26, Curran Associates, Inc. 2013.
13. Barros da Silva A.V., *Data Augmentation for Offline Handwritten Signature Verification*, Recife 2018.
14. Beatrice Drott, Thomas Hassan-Reza, *On-line Handwritten Signature Verification using Machine Learning Techniques with a Deep Learning Approach* [w:] Lund 2015.
15. Bellemare M.G., Candido S., Castro P.S., Gong J., Machado M.C., Moitra S., Ponda S.S., Wang Z., *Autonomous navigation of stratospheric balloons using reinforcement learning*, „Nature” t. 588 nr 7836 (2020), DOI: 10.1038/s41586-020-2939-8.
16. Bennett-Moses L., Chan J., *Algorithmic prediction in policing: assumptions, evaluation, and accountability*, „Policing and Society” t. 28 (2018).
17. Bermejo E., Taniguchi K., Ogawa Y., Martos R., Valsecchi A., Mesejo P., Ibáñez O., Imaizumi K., *Automatic landmark annotation in 3D surface scans of skulls: Methodological proposal and reliability study*, „Computer Methods and Programs in Biomedicine” t. 210 (2021), DOI: 10.1016/j.cmpb.2021.106380.
18. Bibi M., Hamid A., Moetesum M., Siddiqi I., *Document Forgery Detection using Printer Source Identification—A Text-Independent Approach* [w:] *2019 International Conference on Document Analysis and Recognition Workshops (ICDARW)*, t. 8, 2019.
19. Biderman S., Scheirer W.J., *Pitfalls in Machine Learning Research: Reexamining the Development Cycle*, arXiv, 18 sierpnia 2021 r., http://arxiv.org/abs/2011.02832.
20. Bishop C.M., *Neural Networks for Pattern Recognition*, Clarendon Press 1995.
21. Boer H.H. de, Fronczek J., Berger C.E.H., Sjerps M., *The logic of forensic pathology opinion*, „International Journal of Legal Medicine” (2022), DOI: 10.1007/s00414-021-02754-1, https://doi.org/10.1007/s00414-021-02754-1.

22. Bogdal C., Schellenberg R., Lory M., Bovens M., Höpli O., *Recognition of gasoline in fire debris using machine learning: Part II, application of a neural network*, „Forensic Science International" t. 332 (2022), DOI: 10.1016/j.forsciint.2022.111177.
23. Borji A., *Pros and Cons of GAN Evaluation Measures: New Developments*, 2 października 2021 r., http://arxiv.org/abs/2103.09396.
24. Bowyer K.W., King M., Scheirer W., Vangara K., *The Criminality From Face Illusion*, arXiv, 18 listopada 2020 r., http://arxiv.org/abs/2006.03895.
25. Bozza S., Taroni F., Marquis R., Schmittbuhl M., *Probabilistic evaluation of handwriting evidence: likelihood ratio for authorship*, „Applied Statistics" t. 57 (2008).
26. Branting L.K., Pfeifer C., Brown B., Ferro L., Aberdeen J., Weiss B., Pfaff M., Liao B., *Scalable and explainable legal prediction*, „Artificial Intelligence and Law" t. 29 nr 2 (2021), DOI: 10.1007/s10506-020-09273-1.
27. Brink A., Schomaker L., Bulacu M., *Towards Explainable Writer Verification and Identification Using Vantage Writers* [w:] *W: Ninth International Conference on Document Analysis and Recognition (ICDAR 2007*, Washington 2007.
28. Brock A., Donahue J., Simonyan K., *Large Scale GAN Training for High Fidelity Natural Image Synthesis*, 25 lutego 2019 r., http://arxiv.org/abs/1809.11096.
29. Bronsztejn I., Siemiendiajew K., Musiol G., Mühlig H., *Nowoczesne Kompendium Matematyki*, Warszawa 2017.
30. Buchan E., Kelleher L., Clancy M., Stanley Rickard J.J., Oppenheimer P.G., *Spectroscopic molecular-fingerprint profiling of saliva*, „Analytica Chimica Acta" t. 1185 (2021), DOI: 10.1016/j.aca.2021.339074.
31. Buolamwini J., Gebru T., *Gender Shades: Intersectional Accuracy Disparities in Commercial Gender Classification* [w:] *W: Proceedings of the 1st Conference on Fairness, Accountability and Transparency. Proceedings of Machine Learning Research, sine loco*, 2018.
32. Bycroft C., Freeman C., Petkova D., Band G., Elliott L.T., Sharp K., Motyer A., Vukcevic D., Delaneau O., O'Connell J., Cortes A., Welsh S., Young A., Effingham M., McVean G., Leslie S., Allen N., Donnelly P., Marchini J., *The UK*

Biobank resource with deep phenotyping and genomic data, „Nature” t. 562 nr 7726 (2018), DOI: 10.1038/s41586-018-0579-z.

33. Całkiewicz M., *Kryminalistyczne badania patologicznego pisma ręcznego*, Warszawa 2009.
34. Carlini N., Farid H., *Evading Deepfake-Image Detectors with White- and Black-Box Attacks*, arXiv, 1 kwietnia 2020 r., http://arxiv.org/abs/2004.00622.
35. Casali M., Malchiodi D., Spada C., Zanaboni A.M., Cotroneo R., Furci D., Sommariva A., Genovese U., Blandino A., *A pilot study for investigating the feasibility of supervised machine learning approaches for the classification of pedestrians struck by vehicles*, „Journal of Forensic and Legal Medicine” t. 84 (2021), DOI: 10.1016/j.jflm.2021.102256.
36. Castelvecchi D., *Deep learning boosts Google Translate tool*, „Nature” (2016), DOI: 10.1038/nature.2016.20696, https://www.nature.com/articles/nature.2016.20696.
37. Cha S.-H., Tappert C.C., Gibbons M., Chee Y.-M., *Automatic Detection Of Handwriting Forgery Using A Fractal Number Estimate Of Wrinkliness*, „International Journal of Pattern Recognition and Artificial Intelligence” t. 18 nr 07 (2004), DOI: 10.1142/S0218001404003642.
38. Chen W.-F., Ku L.-W., *UTCNN: a Deep Learning Model of Stance Classificationon on Social Media Text*, „arXiv:1611.03599 [cs]” (2016), http://arxiv.org/abs/1611.03599.
39. Chlebowicz P., Łabuz P., Safjański T., *Antykryminalistyka. Taktyka i technika działań kontrwykrywczych*, Warszawa 2022.
40. Choi S., Hill D., Guo L., Nicholas R., Papadopoulos D., Cordeiro M.F., *Automated characterisation of microglia in ageing mice using image processing and supervised machine learning algorithms*, „Scientific Reports” t. 12 nr 1 (2022), DOI: 10.1038/s41598-022-05815-6.
41. Chong E., Han C., Park F.C., *Deep learning networks for stock market analysis and prediction: Methodology, data representations, and case studies*, „Expert Systems with Applications” t. 83 (2017), DOI: 10.1016/j.eswa.2017.04.030.
42. Chugh K., Gupta P., Dhall A., Subramanian R., *Not made for each other- Audio-Visual Dissonance-based Deepfake Detection and Localization* [w:] *Proceedings*

of the 28th ACM International Conference on Multimedia, New York, NY, USA 2020.

43. Ciftci U.A., Demir İ., Yin L., *How Do the Hearts of Deep Fakes Beat? Deep Fake Source Detection via Interpreting Residuals with Biological Signals* [w:] *2020 IEEE International Joint Conference on Biometrics (IJCB)*, Houston, TX, USA 2020.
44. Crawford A., DM A., CP S., *Bayesian hierarchical modelling for the forensic evaluation of handwritten documents*, „Law, Probability and Risk" t. 17 (2020).
45. Csáji B.C., *Approximation with Artificial Neural Networks*, Uniwersytet Eötvös Loránd 2001.
46. Csaky R., *Deep Learning Based Chatbot Models*, „arXiv:1908.08835 [cs]" (2019), http://arxiv.org/abs/1908.08835.
47. Dampage U., Bandaranayake L., Wanasinghe R., Kottahachchi K., Jayasanka B., *Forest fire detection system using wireless sensor networks and machine learning*, „Scientific Reports" t. 12 nr 1 (2022), DOI: 10.1038/s41598-021-03882-9.
48. Dasari A., Prakash S.K.A., Jeni L.A., Tucker C.S., *Evaluation of biases in remote photoplethysmography methods*, „npj Digital Medicine" t. 4 nr 1 (2021), DOI: 10.1038/s41746-021-00462-z.
49. Davis B., Tensmeyer C., Price B., Wigington C., Morse B., Jain R., *Text and Style Conditioned GAN for Generation of Offline Handwriting Lines*, 1 września 2020 r., http://arxiv.org/abs/2009.00678.
50. Davis-Marks I., *Lost Edges of Rembrandt's „Night Watch" Are Restored Using Artificial Intelligence* [na:] „Smithsonian Magazine", https://www.smithsonianmag.com/smart-news/lost-edges-rembrandts-night-watch-are-restored-using-artificial-intelligence-180978056/, dostęp 18 sierpnia 2022 r.
51. Degrave J., Felici F., Buchli J., Neunert M., Tracey B., Carpanese F., Ewalds T., Hafner R., Abdolmaleki A., Casas D. de las, Donner C., Fritz L., Galperti C., Huber A., Keeling J., Tsimpoukelli M., Kay J., Merle A., Moret J.-M., Noury S., Pesamosca F., Pfau D., Sauter O., Sommariva C., Coda S., Duval B., Fasoli A., Kohli P., Kavukcuoglu K., Hassabis D., Riedmiller M., *Magnetic control of*

tokamak plasmas through deep reinforcement learning, „Nature” t. 602 nr 7897 (2022), DOI: 10.1038/s41586-021-04301-9.

52. Del Bimbo A., Cucchiara R., Sclaroff S., Farinella G.M., Mei T., Bertini M., Escalante H.J., Vezzani R. (red.), *Pattern Recognition. ICPR International Workshops and Challenges: Virtual Event, January 10–15, 2021, Proceedings*, t. 1–8, Cham 2021.
53. Dempsey N., Bassed R., Amarasiri R., Blau S., *Exploring the use of machine learning for the assessment of skeletal fracture morphology and differentiation between impact mechanisms: A pilot study*, „Journal of Forensic Sciences” (2022), DOI: 10.1111/1556-4029.14996.
54. Deng L., Suo H., Li D., *Deepfake Video Detection Based on EfficientNet-V2 Network*, „Computational Intelligence and Neuroscience” t. 2022 (2022), DOI: 10.1155/2022/3441549.
55. Dey S., Dutta A., Toledo J.I., Ghosh S.K., Llados J., Pal U., *SigNet: Convolutional Siamese Network for Writer Independent Offline Signature Verification*, „arXiv:1707.02131 [cs]” (2017), http://arxiv.org/abs/1707.02131.
56. Dhariwal P., Jun H., Payne C., Kim J.W., Radford A., Sutskever I., *Jukebox: A Generative Model for Music*, arXiv, 30 kwietnia 2020 r., http://arxiv.org/abs/2005.00341.
57. Diaz M., Ferrer M., Ekladious G., Sabourin R., *Generation of Duplicated Off-Line Signature Images for Verification Systems*, „IEEE Transactions on Pattern Analysis and Machine Intelligence” t. 39 (2016), DOI: 10.1109/TPAMI.2016.2560810.
58. Diaz M., Ferrer M., Sabourin R., *Approaching the Intra-Class Variability in Multi-Script Static Signature Evaluation* [w:] *2016 23rd International Conference on Pattern Recognition (ICPR)*, 2016.
59. Dietterich T.G., *Approximate Statistical Tests for Comparing Supervised Classification Learning Algorithms*, „Neural Computation” t. 10 (1998).
60. Douglas M.R., *Machine learning as a tool in theoretical science*, „Nature Reviews Physics” (2022), DOI: 10.1038/s42254-022-00431-9.

61. Du H., Li P., Zhou H., Gong W., Luo G., Yang P., *WordRecorder: Accurate Acoustic-based Handwriting Recognition Using Deep Learning* [w:] *IEEE INFOCOM 2018 - IEEE Conference on Computer Communications*, 2018.
62. Dumoulin V., Visin F., *A guide to convolution arithmetic for deep learning*, arXiv, 11 stycznia 2018 r., http://arxiv.org/abs/1603.07285.
63. Estai M., Tennant M., Gebauer D., Brostek A., Vignarajan J., Mehdizadeh M., Saha S., *Deep learning for automated detection and numbering of permanent teeth on panoramic images*, „Dento Maxillo Facial Radiology" t. 51 nr 2 (2022), DOI: 10.1259/dmfr.20210296.
64. Etienne H., *The future of online trust (and why Deepfake is advancing it)*, „AI and Ethics" t. 1 nr 4 (2021), DOI: 10.1007/s43681-021-00072-1.
65. Ferrer M.A., Diaz-Cabrera M., Morales A., *Static Signature Synthesis: A Neuromotor Inspired Approach for Biometrics*, „IEEE Transactions on Pattern Analysis and Machine Intelligence" t. 37 nr 3 (2015), DOI: 10.1109/TPAMI.2014.2343981.
66. Fiel S., Sablatnig R., *Writer Identification and Retrieval Using a Convolutional Neural Network* [w:] G. Azzopardi, N. Petkov (red.), *Computer Analysis of Images and Patterns*, Cham 2015.
67. Filipe G., Correia P., Meuwly D., Vloed D., *Empirical validation of likelihood ratio methods; a case study in forensic speaker recognition* [w:] *W: 2016 4th International Conference on Biometrics and Forensics (IWBF). IEEE, Washington*, 2016.
68. Frank J., Eisenhofer T., Schönherr L., Fischer A., Kolossa D., Holz T., *Leveraging Frequency Analysis for Deep Fake Image Recognition*, arXiv, 26 czerwca 2020 r., http://arxiv.org/abs/2003.08685.
69. Fuglsby C., Saunders C.P., *U-statistics for estimating performance metrics in forensic handwriting analysis*, „Journal of Statistical Computation and Simulation" t. 90 nr 6 (2020).
70. Fukushima K., *Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position*, „Biological Cybernetics" t. 36 nr 4 (1980), DOI: 10.1007/BF00344251.

71. Fukushima K., *Cognitron: A self-organizing multilayered neural network*, „Biological Cybernetics" t. 20 nr 3 (1975), DOI: 10.1007/BF00342633.
72. Garain U., Shafait F. (red.), *Computational Forensics: 5th International Workshop, IWCF 2012 Tsukuba, Japan, November 11, 2012 and 6th International Workshop, IWCF 2014 Stockholm, Sweden, August 24, 2014 Revised Selected Papers*, t. 8915, Cham 2015.
73. Garcia-Garcia A., Orts-Escolano S., Oprea S., Villena-Martinez V., Garcia-Rodriguez J., *A Review on Deep Learning Techniques Applied to Semantic Segmentation*, arXiv, 22 kwietnia 2017 r., http://arxiv.org/abs/1704.06857.
74. Garrido Torres J.A., Gharakhanyan V., Artrith N., Eegholm T.H., Urban A., *Augmenting zero-Kelvin quantum mechanics with machine learning for the prediction of chemical reactions at high temperatures*, „Nature Communications" t. 12 nr 1 (2021), DOI: 10.1038/s41467-021-27154-2.
75. Garton N., Ommen D.M., Niemi J., Carriquiry A., *Score-based likelihood ratios to evaluate forensic pattern evidence* [w:] 2020.
76. Ghanim T.M., Nabil A.M., *Offline Signature Verification and Forgery Detection Approach* [w:] *2018 13th International Conference on Computer Engineering and Systems (ICCES)*, Cairo, Egypt 2018.
77. Ghose S., Prevost J.J., *AutoFoley: Artificial Synthesis of Synchronized Sound Tracks for Silent Videos with Deep Learning*, „IEEE Transactions on Multimedia" t. 23 (2021), DOI: 10.1109/TMM.2020.3005033.
78. Gideon S.J., Kandulna A., Kujur A.A., Diana A., Raimond K., *Handwritten Signature Forgery Detection using Convolutional Neural Networks*, „Procedia Computer Science" t. 143 (2018), DOI: 10.1016/j.procs.2018.10.336.
79. Glassner A., *Deep Learning: From Basics to Practice*, Seattle 2018.
80. Goc M., *Współczesny model ekspertyzy pismoznawczej; wykorzystanie nowych metod i technik badawczych*, Warszwa - Szczecin 2015.
81. Goc M., Łuszczuk A., Łuszczuk K., Tomaszewski T., *Programy komputerowe jako narzędzie wspomagające ekspertyzę pisma ręcznego*, „Problemy Kryminalistyki." (2016).
82. Golomingi R., Haas C., Dobay A., Kottner S., Ebert L., *Sperm hunting on optical microscope slides for forensic analysis with deep convolutional networks - a*

feasibility study, „Forensic Science International. Genetics" t. 56 (2022), DOI: 10.1016/j.fsigen.2021.102602.

83. Goodfellow I.J., Pouget-Abadie J., Mirza M., Xu B., Warde-Farley D., Ozair S., Courville A., Bengio Y., *Generative Adversarial Networks*, 10 czerwca 2014 r., http://arxiv.org/abs/1406.2661.
84. Groh M., Epstein Z., Firestone C., Picard R., *Deepfake detection by human crowds, machines, and machine-informed crowds*, „Proceedings of the National Academy of Sciences" t. 119 nr 1 (2022), DOI: 10.1073/pnas.2110013119.
85. Gruza E., Goc M., Moszczyński J., *Kryminalistyka. Czyli o współczesnych metodach dowodzenia przestępstw*, Warszawa 2020.
86. Gu J., Wang X., Li C., Zhao J., Fu W., Liang G., Qiu J., *AI-enabled image fraud in scientific publications*, „Patterns" t. 3 nr 7 (2022), DOI: 10.1016/j.patter.2022.100511.
87. Güera D., Delp E.J., *Deepfake Video Detection Using Recurrent Neural Networks* [w:] *2018 15th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, 2018.
88. Guest D., Cranmer K., Whiteson D., *Deep Learning and its Application to LHC Physics*, „Annual Review of Nuclear and Particle Science" t. 68 nr 1 (2018), DOI: 10.1146/annurev-nucl-101917-021019.
89. Guyon I., Schomaker L., Plamondon R., Liberman M., Janet S., *UNIPEN project of on-line data exchange and recognizer benchmarks* [w:] *W: Proceedings of the 12th IAPR International Conference on Pattern Recognition*, Washington 1994.
90. Hameed M.M., Ahmad R., Kiah M.L.M., Murtaza G., *Machine learning-based offline signature verification systems: A systematic review*, „Signal Processing: Image Communication" t. 93 (2021), DOI: 10.1016/j.image.2021.116139.
91. Han M., Du S., Ge Y., Zhang D., Chi Y., Long H., Yang J., Yang Y., Xin J., Chen T., Zheng N., Guo Y., *With or without human interference for precise age estimation based on machine learning?*, „International Journal of Legal Medicine" (2022), DOI: 10.1007/s00414-022-02796-z, https://doi.org/10.1007/s00414-022-02796-z.
92. Haraksim R., Meuwly D., Vergeer P., *Fingerprint Evidence Evaluation, Robustness to the Lack of Data* [w:] Netherlands Forensic Institute, Hague 2012.

93. Haraksim R., Ramos D., Meuwly D., *Validation of likelihood ratio methods for forensic evidence evaluation handling multimodal score distributions*, „IET Biometrics" t. 6 (2017).
94. Haraksim R., Ramos D., Meuwly D., Berger C.E.H., *Measuring coherence of computer-assisted likelihood ratio methods*, „Forensic Science International" t. 249 (2015).
95. Hashmi M.F., Ashish B.K.K., Keskar A.G., Bokde N.D., Yoon J.H., Geem Z.W., *An Exploratory Analysis on Visual Counterfeits Using Conv-LSTM Hybrid Architecture*, „IEEE Access" t. 8 (2020), DOI: 10.1109/ACCESS.2020.2998330.
96. Hassan M., Wang Y., Wang D., Li D., Liang Y., Zhou Y., Xu D., *Deep learning analysis and age prediction from shoeprints*, „Forensic Science International" t. 327 (2021), DOI: 10.1016/j.forsciint.2021.110987.
97. Haykin S., *Neural Networks: A Comprehensive Foundation*, Prentice Hall 1999.
98. He Z.-Y., Chen Q.-H., Chen D.-F., *A neural network expert system for Chinese handwriting-based writer identification* [w:] *Proceedings. International Conference on Machine Learning and Cybernetics*, t. 4, 2002.
99. Hebb D., *The Organization of Behavior: A Neuropsychological Theory*, New York 1949.
100. Hochreiter S., Schmidhuber J., *Long Short-Term Memory*, „Neural Computation" t. 9 nr 8 (1997), DOI: 10.1162/neco.1997.9.8.1735.
101. Hopfield J.J., *Neural networks and physical systems with emergent collective computational abilities.*, „Proceedings of the National Academy of Sciences" t. 79 nr 8 (1982), DOI: 10.1073/pnas.79.8.2554.
102. House of Commons Science and Technology Committee, *Forensic Science on Trial, Seventh Report of Session 2004–05, HC 96–I.*, London 2005.
103. Hu L., Xing Y., Jiang P., Gan L., Zhao F., Peng W., Li W., Tong Y., Deng S., *Predicting the postmortem interval using human intestinal microbiome data and random forest algorithm*, „Science & Justice" t. 61 nr 5 (2021), DOI: 10.1016/j.scijus.2021.06.006.
104. Hu Y.H., Hwang J.-N., *Handbook of neural network signal processing*, Boca Raton 2002.

105. Hubel D.H., Wiesel T.N., *Receptive fields and functional architecture of monkey striate cortex*, „The Journal of Physiology" t. 195 nr 1 (1968), DOI: 10.1113/jphysiol.1968.sp008455.
106. Hubel D.H., Wiesel T.N., *Receptive fields of single neurones in the cat's striate cortex*, „The Journal of Physiology" t. 148 nr 3 (1959).
107. Huber R.A., Headrick A.M., Harralson H.H., Miler L.S., *Handwriting Identification Facts and Fundamentals*, Boca Raton 2018.
108. Hussain S., Neekhara P., Jere M., Koushanfar F., McAuley J., *Adversarial Deepfakes: Evaluating Vulnerability of Deepfake Detectors to Adversarial Examples*, arXiv, 7 listopada 2020 r., http://arxiv.org/abs/2002.12749.
109. Ibanez V., Gunz S., Erne S., Rawdon E.J., Ampanozi G., Franckenberg S., Sieberth T., Affolter R., Ebert L.C., Dobay A., *RiFNet: Automated rib fracture detection in postmortem computed tomography*, „Forensic Science, Medicine, and Pathology" (2021), DOI: 10.1007/s12024-021-00431-8.
110. Instytut Ekspertyz Sądowych w Krakowie, *Słownik Terminów Pismoznawczych* [na:] http://prawouam-stp.home.amu.edu.pl/, 2007 r., dostęp 20 września 2021 r.
111. Intasuwan P., Palee P., Sinthubua A., Mahakkanukrauh P., *Comparison of sex determination using three methods applied to the greater sciatic notch of os coxae in a Thai population: Dry bone morphology, 2-dimensional photograph morphometry, and deep learning artificial neural network*, „Medicine, Science and the Law" (2022), DOI: 10.1177/00258024221079092.
112. International Association for Pattern Recognition, *International Conference on Document Analysis and Recognition 2021; Competition on On-line Signature Verification; SVC 2021* [na:] https://sites.google.com/view/SVC2021/home, dostęp 20 września 2021 r.
113. Ioffe S., Szegedy C., *Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift*, „arXiv:1502.03167 [cs]" (2015), http://arxiv.org/abs/1502.03167.
114. Ismail A., Elpeltagy M., Zaki M., ElDahshan K.A., *Deepfake video detection: YOLO-Face convolution recurrent approach*, „PeerJ Computer Science" t. 7 (2021), DOI: 10.7717/peerj-cs.730.

115. Isola P., Zhu J.-Y., Zhou T., Efros A.A., *Image-to-Image Translation with Conditional Adversarial Networks*, 26 listopada 2018 r., http://arxiv.org/abs/1611.07004.
116. J. Ormond, *Fathers of the Deep Learning revolution receive 2018 ACM A.M. Turing Award* [na:] https://www.acm.org/media-center/2019/march/turing-award-2018, dostęp 27 stycznia 2023 r.
117. Jaiswal G., Sharma A., Yadav S.K., *Deep feature extraction for document forgery detection with convolutional autoencoders*, „Computers & Electrical Engineering" t. 99 (2022), DOI: 10.1016/j.compeleceng.2022.107770.
118. Janocha K., Czarnecki W.M., *On Loss Functions for Deep Neural Networks in Classification*, arXiv, 18 lutego 2017 r., http://arxiv.org/abs/1702.05659.
119. Jarrett K., Kavukcuoglu K., Ranzato M., LeCun Y., *What is the best multi-stage architecture for object recognition?* [w:] *2009 IEEE 12th International Conference on Computer Vision*, 2009.
120. Javed A.R., Jalil Z., Zehra W., Gadekallu T.R., Suh D.Y., Piran Md.J., *A comprehensive survey on digital video forensics: Taxonomy, challenges, and future directions*, „Engineering Applications of Artificial Intelligence" t. 106 (2021), DOI: 10.1016/j.engappai.2021.104456.
121. Jin X., Li Z., Liu K., Zou D., Li X., Zhu X., Zhou Z., Sun Q., Liu Q., *Focusing on Persons: Colorizing Old Images Learning from Modern Historical Movies*, „arXiv:2108.06515 [cs]" (2021), http://arxiv.org/abs/2108.06515.
122. Jumper J., Evans R., Pritzel A., Green T., Figurnov M., Ronneberger O., Tunyasuvunakool K., Bates R., Žídek A., Potapenko A., Bridgland A., Meyer C., Kohl S.A.A., Ballard A.J., Cowie A., Romera-Paredes B., Nikolov S., Jain R., Adler J., Back T., Petersen S., Reiman D., Clancy E., Zielinski M., Steinegger M., Pacholska M., Berghammer T., Bodenstein S., Silver D., Vinyals O., Senior A.W., Kavukcuoglu K., Kohli P., Hassabis D., *Highly accurate protein structure prediction with AlphaFold*, „Nature" t. 596 nr 7873 (2021), DOI: 10.1038/s41586-021-03819-2.
123. Jung T., Kim S., Kim K., *DeepVision: Deepfakes Detection Using Human Eye Blinking Pattern*, „IEEE Access" t. 8 (2020), DOI: 10.1109/ACCESS.2020.2988660.

124. Kalera M.K., Srihari S., Xu A., *Offline signature verification and identification using distance statistics*, „International Journal of Pattern Recognition and Artificial Intelligence" t. 18 nr 07 (2004), DOI: 10.1142/S0218001404003630.
125. Kamnitsas K., Ledig C., Newcombe V.F.J., Simpson J.P., Kane A.D., Menon D.K., Rueckert D., Glocker B., *Efficient Multi-Scale 3D CNN with Fully Connected CRF for Accurate Brain Lesion Segmentation*, „Medical Image Analysis" t. 36 (2017), DOI: 10.1016/j.media.2016.10.004.
126. Kang L., Riba P., Wang Y., Rusiñol M., Fornés A., Villegas M., *GANwriting: Content-Conditioned Generation of Styled Handwritten Word Images*, 21 lipca 2020 r., http://arxiv.org/abs/2003.02567.
127. Kingma D.P., Ba J., *Adam: A Method for Stochastic Optimization*, arXiv, 29 stycznia 2017 r., http://arxiv.org/abs/1412.6980.
128. Kleber F., Fiel S., Diem M., Sablatnig R., *CVL-DataBase: An Off-Line Database for Writer Retrieval, Writer Identification and Word Spotting* [w:] *2013 12th International Conference on Document Analysis and Recognition*, Washington, DC, USA 2013.
129. Klingberg J., Keen B., Cawley A., Pasin D., Fu S., *Developments in high-resolution mass spectrometric analyses of new psychoactive substances*, „Archives of Toxicology" (2022), DOI: 10.1007/s00204-022-03224-2, https://doi.org/10.1007/s00204-022-03224-2.
130. Konieczny K., Widła T., Widacki J., *Kryminalistyka*, Warszawa 2016.
131. Krizhevsky A., *Convolutional Deep Belief Networks on CIFAR-10*, 2010 r., https://www.semanticscholar.org/paper/Convolutional-Deep-Belief-Networks-on-CIFAR-10-Krizhevsky/bea5780d621e669e8069f05d0f2fc0db9df4b50f.
132. Krizhevsky A., Sutskever I., Hinton G.E., *ImageNet Classification with Deep Convolutional Neural Networks* [w:] *Advances in Neural Information Processing Systems*, t. 25, Curran Associates, Inc. 2012.
133. Kudeikina I., Loseviča M., Gutorova N.O., *Legal and practical problems of use of artificial intelligence-based robots in forensic psychiatry*, „Wiadomosci Lekarskie (Warsaw, Poland: 1960)" t. 74 nr 11 cz 2 (2021).

134. Kulesh V., Schaffer K., Sethi I., Schwartz M., *Handwriting Quality Evaluation* [w:] S. Singh, N. Murshed, W. Kropatsch (red.), *Advances in Pattern Recognition — ICAPR 2001*, t. 2013, Berlin, Heidelberg 2001.

135. Lagemann C., Lagemann K., Mukherjee S., Schröder W., *Deep recurrent optical flow learning for particle image velocimetry data*, „Nature Machine Intelligence" t. 3 nr 7 (2021), DOI: 10.1038/s42256-021-00369-0.

136. Lai S., Jin L., Zhu Y., Li Z., Lin L., *SynSig2Vec: Forgery-free Learning of Dynamic Signature Representations by Sigma Lognormal-based Synthesis*, „IEEE Transactions on Pattern Analysis and Machine Intelligence" (2021), DOI: 10.1109/TPAMI.2021.3087619.

137. LeCun Y., Boser B., Denker J., Henderson D., Howard R., Hubbard W., Jackel L., *Handwritten Digit Recognition with a Back-Propagation Network* [w:] *Advances in Neural Information Processing Systems*, t. 2, Morgan-Kaufmann 1989.

138. Lecun Y., Bottou L., Bengio Y., Haffner P., *Gradient-based learning applied to document recognition*, „Proceedings of the IEEE" t. 86 nr 11 (1998), DOI: 10.1109/5.726791.

139. Ledig C., Theis L., Huszar F., Caballero J., Cunningham A., Acosta A., Aitken A., Tejani A., Totz J., Wang Z., Shi W., *Photo-Realistic Single Image Super-Resolution Using a Generative Adversarial Network*, „arXiv:1609.04802 [cs, stat]" (2017), http://arxiv.org/abs/1609.04802.

140. Lee H., Grosse R., Ranganath R., Ng A.Y., *Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations* [w:] *Proceedings of the 26th Annual International Conference on Machine Learning*, New York, NY, USA 2009.

141. Lee S.Y., Lee S.T., Suh S., Ko B.J., Oh H.B., *Revealing Unknown Controlled Substances and New Psychoactive Substances Using High-Resolution LC-MS/MS Machine Learning Models and the Hybrid Similarity Search Algorithm*, „Journal of Analytical Toxicology" (2021), DOI: 10.1093/jat/bkab098.

142. Li H., Fang S., Mukhopadhyay S., Saykin A.J., Shen L., *Interactive Machine Learning by Visualization: A Small Data Solution*, „Proceedings : ... IEEE

International Conference on Big Data. IEEE International Conference on Big Data" t. 2018 (2018), DOI: 10.1109/BigData.2018.8621952.

143. Li Y., *Deep Reinforcement Learning: An Overview*, arXiv, 25 listopada 2018 r., http://arxiv.org/abs/1701.07274.
144. Li Y., Chang M.-C., Lyu S., *In Ictu Oculi: Exposing AI Created Fake Videos by Detecting Eye Blinking* [w:] 2018.
145. Li Y., Lyu S., *Exposing DeepFake Videos By Detecting Face Warping Artifacts*, „CVPR Workshops" (2019).
146. Li Y., Yang X., Sun P., Qi H., Lyu S., *Celeb-DF: A Large-scale Challenging Dataset for DeepFake Forensics*, arXiv, 16 marca 2020 r., http://arxiv.org/abs/1909.12962.
147. Li Y., Yang X., Wu B., Lyu S., *Hiding Faces in Plain Sight: Disrupting AI Face Synthesis with Adversarial Perturbations*, arXiv, 21 czerwca 2019 r., http://arxiv.org/abs/1906.09288.
148. Liu Z., Qi X., Torr P.H.S., *Global Texture Enhancement for Fake Face Detection in the Wild* [w:] *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Seattle, WA, USA 2020.
149. Löwel S., Singer W., *Selection of Intrinsic Horizontal Connections in the Visual Cortex by Correlated Neuronal Activity*, „Science" t. 255 nr 5041 (1992), DOI: 10.1126/science.1372754.
150. Lucic M., Kurach K., Michalski M., Gelly S., Bousquet O., *Are GANs Created Equal? A Large-Scale Study*, arXiv, 29 października 2018 r., http://arxiv.org/abs/1711.10337.
151. Lutz K., Bassett R., *DeepFake Detection with Inconsistent Head Poses: Reproducibility and Analysis*, „ArXiv" (2021).
152. Mameli F., Bertini M., Galteri L., Del Bimbo A., *A NoGAN approach for image and video restoration and compression artifact removal* [w:] *2020 25th International Conference on Pattern Recognition (ICPR)*, Milan, Italy 2021.
153. Marcon F., Pasquini C., Boato G., *Detection of Manipulated Face Videos over Social Networks: A Large-Scale Study*, „Journal of Imaging" t. 7 nr 10 (2021), DOI: 10.3390/jimaging7100193.

154. Marti U., Bunke H., *The IAM-database: An English Sentence Database for Off-line Handwriting Recognition*, „International Journal on Document Analysis and Recognition" t. 5 (2002).
155. McGreevy N., *Hear an A.I.-Generated Andy Warhol „Read" His Diary to You in New Documentary* [na:] „Smithsonian Magazine", https://www.smithsonianmag.com/smart-news/an-ai-generated-andy-warhol-reads-his-diary-to-you-in-new-documentary-180979658/, dostęp 18 sierpnia 2022 r.
156. Medsker L., Jain L.C., *Recurrent Neural Networks: Design and Applications*, CRC Press 1999.
157. Miłkowski M., Hensel W.M., Hohoł M., *Replicability or reproducibility? On the replication crisis in computational neuroscience and sharing only relevant detail*, „Journal of Computational Neuroscience" t. 45 (2018).
158. Minsky M., Papert S., *Perceptrons; an Introduction to Computational Geometry*, MIT Press 1969.
159. Mirsky Y., Lee W., *The Creation and Detection of Deepfakes: A Survey*, 31 stycznia 2022 r., http://arxiv.org/abs/2004.11138.
160. Mordvintsev A., Olah C., Schubert L., *Feature Visualization; How neural networks build up their understanding of images*, „Distill" (2017), DOI: 10.23915/distill.00007.
161. Öhman C., *Introducing the pervert's dilemma: a contribution to the critique of Deepfake Pornography*, „Ethics and Information Technology" t. 22 nr 2 (2020), DOI: 10.1007/s10676-019-09522-1.
162. Olshausen B.A., *Linear Hebbian learning and PCA; Redwood Center for Theoretical Neuroscience at the University of California in Berkeley* [na:] https://redwood.berkeley.edu/wp-content/uploads/2018/08/handout-hebb-PCA.pdf, 2012 r., dostęp 3 listopada 2022 r.
163. Ommen D.M., Saunders C.P., *Building a unified statistical framework for the forensic identification of source problems*, „Law, Probability and Risk" t. 17 nr 2 (2018), DOI: 10.1093/lpr/mgy008.
164. *Optimizing the synergy between physics and machine learning*, „Nature Machine Intelligence" t. 3 nr 11 (2021), DOI: 10.1038/s42256-021-00416-w.

165. Oura P., Junno A., Junno J.-A., *Deep learning in forensic shotgun pattern interpretation – A proof-of-concept study*, „Legal Medicine" t. 53 (2021), DOI: 10.1016/j.legalmed.2021.101960.
166. Øygard A., *Visualizing GoogLeNet Classes* [na:] https://www.auduno.com/2015/07/29/visualizing-googlenet-classes/, 2015 r., dostęp 14 marca 2022 r.
167. Papernot N., McDaniel P., Goodfellow I., *Transferability in Machine Learning: from Phenomena to Black-Box Attacks using Adversarial Samples*, arXiv, 23 maja 2016 r., http://arxiv.org/abs/1605.07277.
168. Park T., Liu M.-Y., Wang T.-C., Zhu J.-Y., *Semantic Image Synthesis with Spatially-Adaptive Normalization*, arXiv, 5 listopada 2019 r., http://arxiv.org/abs/1903.07291.
169. Pascanu R., Mikolov T., Bengio Y., *On the difficulty of training Recurrent Neural Networks*, arXiv, 15 lutego 2013 r., http://arxiv.org/abs/1211.5063.
170. Pastor-Pellicer J., Castro-Bleda M.J., España-Boquera S., Zamora-Martínez F., *Handwriting recognition by using deep learning to extract meaningful features*, „AI Communications" t. 32 nr 2 (2019), DOI: 10.3233/AIC-170562.
171. Pawelec M., *Deepfakes and Democracy (Theory): How Synthetic Audio-Visual Media for Disinformation and Hate Speech Threaten Core Democratic Functions*, „Digital Society" t. 1 nr 2 (2022), DOI: 10.1007/s44206-022-00010-6.
172. Penrose R., *Droga do Rzeczywistości*, Warszwa 2006.
173. Pinto N., Doukhan D., DiCarlo J.J., Cox D.D., *A high-throughput screening approach to discovering good forms of biologically inspired visual representation*, „PLoS computational biology" t. 5 nr 11 (2009), DOI: 10.1371/journal.pcbi.1000579.
174. Popper K., *Logika odkrycia naukowego*, Warszawa 2002.
175. Popper K., *The Logic of Scientific Discovery*, London 2002.
176. Pośpiech E., Teisseyre P., Mielniczuk J., Branicki W., *Predicting Physical Appearance from DNA Data—Towards Genomic Solutions*, „Genes" t. 13 nr 1 (2022), DOI: 10.3390/genes13010121.

177. Poterek Q., Herrault P.-A., Skupinski G., Sheeren D., *Deep Learning for Automatic Colorization of Legacy Grayscale Aerial Photographs*, „IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing" t. 13 (2020), DOI: 10.1109/JSTARS.2020.2992082.
178. Publications Office of the European Union, *Follow the steps of procedure 2021/0106/COD* [na:] https://eur-lex.europa.eu/legal-content/EN/HIS/?uri=celex:52021PC0206, dostęp 6 marca 2023 r.
179. Qi H., Guo Q., Juefei-Xu F., Xie X., Ma L., Feng W., Liu Y., Zhao J., *DeepRhythm: Exposing DeepFakes with Attentional Visual Heartbeat Rhythms*, „arXiv:2006.07634 [cs]" (2020), http://arxiv.org/abs/2006.07634.
180. *QLED 8K: Where AI Upscaling Meets Deep Learning* [na:] https://news.samsung.com/global/the-future-of-viewing-1-qled-8k-where-ai-upscaling-meets-deep-learning, dostęp 18 sierpnia 2022 r.
181. Ramachandran P., Zoph B., Le Q.V., *Searching for Activation Functions*, arXiv, 27 października 2017 r., http://arxiv.org/abs/1710.05941.
182. Ramesh A., Dhariwal P., Nichol A., Chu C., Chen M., *Hierarchical Text-Conditional Image Generation with CLIP Latents*, arXiv, 12 kwietnia 2022 r., http://arxiv.org/abs/2204.06125.
183. Ramesh A., Pavlov M., Goh G., Gray S., Voss C., Radford A., Chen M., Sutskever I., *Zero-Shot Text-to-Image Generation*, arXiv, 26 lutego 2021 r., http://arxiv.org/abs/2102.12092.
184. Ramos D., Gonzalez-Rodriguez J., *Reliable support: Measuring calibration of likelihood ratios*, „Forensic Science International" t. 230 (2013).
185. Ramos D., Haraksim R., Meuwly D., *Likelihood ratio data to report the validation of a forensic fingerprint evaluation method*, „Data in Brief" t. 10 (2017).
186. Rao Q., Frtunikj J., *Deep learning for self-driving cars: chances and challenges* [w:] *Proceedings of the 1st International Workshop on Software Engineering for AI in Autonomous Systems*, New York, NY, USA 2018.
187. Ravuri S., Lenc K., Willson M., Kangin D., Lam R., Mirowski P., Fitzsimons M., Athanassiadou M., Kashem S., Madge S., Prudden R., Mandhane A., Clark A., Brock A., Simonyan K., Hadsell R., Robinson N., Clancy E., Arribas A.,

Mohamed S., *Skilful precipitation nowcasting using deep generative models of radar*, „Nature” t. 597 nr 7878 (2021), DOI: 10.1038/s41586-021-03854-z.

188. Richardson R., Schultz J.M., Crawford K., *Dirty Data, Bad Predictions: How Civil Rights Violations Impact Police Data, Predictive Policing Systems, and Justice*, „New York University Law Review” t. 94 (2020).
189. Rojas R., *Neural Networks: A Systematic Introduction*, Springer Science & Business Media 2013.
190. Ronneberger O., Fischer P., Brox T., *U-Net: Convolutional Networks for Biomedical Image Segmentation*, arXiv, 18 maja 2015 r., http://arxiv.org/abs/1505.04597.
191. Rosenblatt F., *The perceptron: A probabilistic model for information storage and organization in the brain*, „Psychological Review” t. 65 (1958), DOI: 10.1037/h0042519.
192. Roy P., Bag S., *Detection of Handwritten Document Forgery by Analyzing Writers' Handwritings* [w:] B. Deka, P. Maji, S. Mitra, D. K. Bhattacharyya, P. K. Bora, S. K. Pal (red.), *Pattern Recognition and Machine Intelligence*, Cham 2019.
193. Rudin C., *Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead*, „Nature Machine Intelligence” t. 1 nr 5 (2019), DOI: 10.1038/s42256-019-0048-x.
194. Rumelhart D.E., Hinton G.E., Williams R.J., *Learning representations by back-propagating errors*, „Nature” t. 323 nr 6088 (1986), DOI: 10.1038/323533a0.
195. Ryu S.-J., Lee H.-Y., Cho I.-W., Lee H.-K., *Document Forgery Detection with SVM Classifier and Image Quality Measures* [w:] Y.-M. R. Huang, C. Xu, K.-S. Cheng, J.-F. K. Yang, M. N. S. Swamy, S. Li, J.-W. Ding (red.), *Advances in Multimedia Information Processing - PCM 2008*, Berlin, Heidelberg 2008.
196. Sadowski P., *Notes on Backpropagation; University of California Irvine* [na:] https://www.ics.uci.edu/~pjsadows/notes.pdf, dostęp 28 listopada 2022 r.
197. Sako H., Franke K.Y., Saitoh S. (red.), *Computational Forensics: 4th International Workshop, IWCF 2010 Tokyo, Japan, November 11-12, 2010 Revised Selected Papers*, t. 6540, Berlin, Heidelberg 2011.
198. Santosh K.C., Pradeep N., Goel V., Ranjan R., Pandey E., Shukla P.K., Nuagah S.J., *Machine Learning Techniques for Human Age and Gender Identification*

Based on Teeth X-Ray Images, „Journal of Healthcare Engineering" t. 2022 (2022), DOI: 10.1155/2022/8302674.

199. Schmidhuber J., *Deep Learning in Neural Networks: An Overview*, „Neural Networks" t. 61 (2015), DOI: 10.1016/j.neunet.2014.09.003.
200. Schreyer M., Sattarov T., Reimer B., Borth D., *Adversarial Learning of Deepfakes in Accounting*, arXiv, 9 października 2019 r., http://arxiv.org/abs/1910.03810.
201. Schroff F., Kalenichenko D., Philbin J., *FaceNet: A Unified Embedding for Face Recognition and Clustering* [w:] *W: Conference on Computer Vision and Pattern Recognition. IEEE, Piscataway*, 2015.
202. Shan S., Wenger E., Zhang J., Li H., Zheng H., Zhao B.Y., *Fawkes: protecting privacy against unauthorized deep learning models* [w:] *Proceedings of the 29th USENIX Conference on Security Symposium*, USA 2020.
203. Sidheekh S., Aimen A., Krishnan N.C., *On Characterizing GAN Convergence Through Proximal Duality Gap* [w:] *Proceedings of the 38th International Conference on Machine Learning*, PMLR 2021.
204. Silver D., Hubert T., Schrittwieser J., Antonoglou I., Lai M., Guez A., Lanctot M., Sifre L., Kumaran D., Graepel T., Lillicrap T., Simonyan K., Hassabis D., *Mastering Chess and Shogi by Self-Play with a General Reinforcement Learning Algorithm*, „arXiv:1712.01815 [cs]" (2017), http://arxiv.org/abs/1712.01815.
205. Silver D., Schrittwieser J., Simonyan K., Antonoglou I., Huang A., Guez A., Hubert T., Baker L., Lai M., Bolton A., Chen Y., Lillicrap T., Hui F., Sifre L., Driessche G. van den, Graepel T., Hassabis D., *Mastering the game of Go without human knowledge*, „Nature" t. 550 nr 7676 (2017), DOI: 10.1038/nature24270.
206. Simonyan K., Zisserman A., *Very Deep Convolutional Networks for Large-Scale Image Recognition*, 10 kwietnia 2015 r., http://arxiv.org/abs/1409.1556.
207. Srihari S., Shin Y.-C., Lee S., Govindaraju V., Cha S.-H., Tomai C.I., Zhang B., Shekhawat A., Bartnik D., Yang W., Setlur S., Kilinskas P., Kunderman F., Liu X., Shi Z., Ramanaprasad V., *Method and apparatus for analyzing and/or comparing handwritten and/or biometric samples. United States Patent US7580551B1, filed 30 June 2003, and issued 25 August 2009* [na:] https://patents.google.com/patent/US7580551/en, dostęp 13 marca 2023 r.

208. Srihari S.N., Franke K. (red.), *Computational Forensics: Second International Workshop, IWCF 2008, Washington, DC, USA, August 7-8, 2008. Proceedings*, t. 5158, Berlin, Heidelberg 2008.

209. Srinivasan H., Srihari S.N., Beal M.J., *Machine Learning for Signature Verification* [w:] P. K. Kalra, S. Peleg (red.), *Computer Vision, Graphics and Image Processing*, Berlin, Heidelberg 2006.

210. Steffensmeier D., *Age, Gender, and Crime Across Three Historical Periods: 1935, 1960, and 1985*, „Social Forces" t. 69 (1991).

211. Strom R.W., *Hebbian Learning in Multilayer Neural Networks* [w:] Los Angeles 2007.

212. Sudana O., Gunaya I.W., Darma Putra I.K.G., *Handwriting identification using deep convolutional neural network method*, „TELKOMNIKA (Telecommunication Computing Electronics and Control)" t. 18 nr 4 (2020), DOI: 10.12928/telkomnika.v18i4.14864.

213. Sutton R.S., Barto A.G., *Reinforcement Learning: An Introduction*, Cambridge 2015.

214. Suwajanakorn S., Seitz S.M., Kemelmacher-Shlizerman I., *Synthesizing Obama: learning lip sync from audio*, „ACM Transactions on Graphics" t. 36 nr 4 (2017), DOI: 10.1145/3072959.3073640.

215. Szostek D., *Is the Traditional Method of Regulation (the Legislative Act) Sufficient to Regulate Artificial Intelligence, or Should It Also Be Regulated by an Algorithmic Code?*, „Białostockie Studia Prawnicze" t. 26 nr 3 (2021), DOI: 10.15290/bsp.2021.26.03.03.

216. Tadeusiewicz R., *Sieci Neuronowe*, Warszawa 1993.

217. Taroni F., Bozza S., Biedermann A., Garbolino P., Aitken C., *Data Analysis in Forensic Science; A Bayesian Decision Perspective*, Chichester 2010.

218. Tauseef A., Spreeuwers L., Veldhuis R., Meuwly D., *Biometric evidence evaluation: an empirical assessment of the effect of different training data*, „IET Biometrics" t. 3 (2014).

219. Taylor M., Bird C., Bishop B., Burkes T., Caligiuri M.P., Found B., Grose W.P., Logan L.R., Melson K.E., Merlino M.L., Miller L.S., Mohammed L., Morris J., Osborn J.P., Osborne N., Ostrum B., Saunders C.P., Shappell S.A., Sheets H.D.,

Srihari S.N., Stoel R.D., Vastrick T.W., Waltke H.E., Will E.J., *Forensic Handwriting Examination and Human Factors: Improving the Practice Through a Systems Approach*, Gaithersburg 2020.

220. Thanh-Tung H., Tran T., *On Catastrophic Forgetting and Mode Collapse in Generative Adversarial Networks*, arXiv, 21 marca 2020 r., http://arxiv.org/abs/1807.04015.
221. The TensorFlow GAN Authors, *Losses that are useful for training GANs* [na:] https://github.com/tensorflow/gan/blob/master/tensorflow_gan/python/losses/losses_impl.py, dostęp 27 września 2022 r.
222. Tomaszewska-Michalak M., *Fake news – wstępna analiza zjawiska*, „Przegląd Politologiczny" nr 1 (2021), DOI: 10.14746/pp.2021.26.1.4.
223. Tomaszewski T., *Jeszcze o tym, czy warto badać kopie i inne wtórne odwzorowania pisma ręcznego*, „Człowiek i Dokumenty" nr 42 (2016).
224. Tomaszewski T., *Dowód z opinii biegłego w procesie karnym*, Kraków 2000.
225. Trigueros D.S., Meng L., Hartnett M., *Face Recognition: From Traditional to Deep Learning Methods*, „arXiv:1811.00116 [cs]" (2018), http://arxiv.org/abs/1811.00116.
226. Tulder G. van, *Elastic deformations for N-dimensional images (Python, SciPy, NumPy, TensorFlow, PyTorch)* [na:] https://pypi.org/project/elasticdeform/, dostęp 27 września 2022 r.
227. Turaga S.C., Murray J.F., Jain V., Roth F., Helmstaedter M., Briggman K., Denk W., Seung H.S., *Convolutional networks can learn to generate affinity graphs for image segmentation*, „Neural computation" t. 22 nr 2 (2010), DOI: 10.1162/neco.2009.10-08-881.
228. United States Supreme Court, *Daubert v. Merrell Dow Pharmaceuticals Inc., 509 U.S. 579* [w:] F. D. Wagner (red.), *United States Reports Volume 509, Cases Adjudged in the Supreme Court At October Term 1992*, Washington 1997.
229. V K., S S.P., *Hybrid machine learning classification scheme for speaker identification*, „Journal of Forensic Sciences" (2022), DOI: 10.1111/1556-4029.15006.

230. Vaccari C., Chadwick A., *Deepfakes and Disinformation: Exploring the Impact of Synthetic Political Video on Deception, Uncertainty, and Trust in News*, „Social Media + Society" t. 6 nr 1 (2020), DOI: 10.1177/2056305120903408.

231. Vorugunti C.S., Pulabaigari V., Gorthi R.K.S.S., Mukherjee P., *OSVFuseNet: Online Signature Verification by feature fusion and depth-wise separable convolution based deep learning*, „Neurocomputing" t. 409 (2020), DOI: 10.1016/j.neucom.2020.05.072.

232. Wang Z., Guo Y., Zuo W., *Deepfake Forensics via an Adversarial Game*, „IEEE Transactions on Image Processing" t. 31 (2022), DOI: 10.1109/TIP.2022.3172845.

233. Weng L., *From GAN to WGAN*, arXiv, 18 kwietnia 2019 r., http://arxiv.org/abs/1904.08994.

234. Werbos P.J., *Applications of advances in nonlinear sensitivity analysis* [w:] R. F. Drenick, F. Kozin (red.), *System Modeling and Optimization*, Berlin, Heidelberg 1982.

235. Widła T., *Metodyka Ekspertyzy* [w:] M. Kała, D. Wild, J. Wójcikiewicz (red.), *Ekspertyza sądowa: zagadnienia wybrane*, Warszawa 2017.

236. Willett F.R., Avansino D.T., Hochberg L.R., Henderson J.M., Shenoy K.V., *High-performance brain-to-text communication via handwriting*, „Nature" t. 593 nr 7858 (2021), DOI: 10.1038/s41586-021-03506-2.

237. Wniosek dotyczący rozporządzenia Parlamentu Europejskiego i Rady (UE) nr COM/2021/206 final z dnia 21 kwietnia 2021 r. ustanawiającego zharmonizowane przepisy dotyczące sztucznej inteligencji (akt w sprawie sztucznej inteligencji) i zmieniającego niektóre akty ustawodawcze unii.

238. Wu X., Zhang X., *Responses to Critiques on Machine Learning of Criminality Perceptions (Addendum of arXiv:1611.04135)*, arXiv, 26 maja 2017 r., http://arxiv.org/abs/1611.04135.

239. Xie C., Zhuang X.-X., Niu Z., Ai R., Lautrup S., Zheng S., Jiang Y., Han R., Gupta T.S., Cao S., Lagartos-Donate M.J., Cai C.-Z., Xie L.-M., Caponio D., Wang W.-W., Schmauck-Medina T., Zhang J., Wang H., Lou G., Xiao X., Zheng W., Palikaras K., Yang G., Caldwell K.A., Caldwell G.A., Shen H.-M., Nilsen H., Lu J.-H., Fang E.F., *Amelioration of Alzheimer's disease pathology by mitophagy*

inducers identified via machine learning and a cross-species workflow, „Nature Biomedical Engineering” t. 6 nr 1 (2022), DOI: 10.1038/s41551-021-00819-5.

240. Yang P., *Dual-Domain Fusion Convolutional Neural Network for Contrast Enhancement Forensics*, „Entropy (Basel, Switzerland)” t. 23 nr 10 (2021), DOI: 10.3390/e23101318.
241. Yu N., Davis L., Fritz M., *Attributing Fake Images to GANs: Learning and Analyzing GAN Fingerprints*, 16 sierpnia 2019 r., http://arxiv.org/abs/1811.08180.
242. Yu P., Xia Z., Fei J., Lu Y., *A Survey on Deepfake Video Detection*, „IET Biometrics” t. 10 nr 6 (2021), DOI: 10.1049/bme2.12031.
243. Yu Z., Qin Y., Li X., Zhao C., Lei Z., Zhao G., *Deep Learning for Face Anti-Spoofing: A Survey*, arXiv, 2 września 2022 r., http://arxiv.org/abs/2106.14948.
244. Załączniki do wniosku dotyczącego rozporządzenia Parlamentu Europejskiego i Rady (UE) nr COM/2021/206 final z dnia 21 kwietnia 2021 r. ustanawiającego zharmonizowane przepisy dotyczące sztucznej inteligencji (akt w sprawie sztucznej inteligencji) i zmieniającego niektóre akty ustawodawcze unii.
245. Zhang C., Bengio S., Hardt M., Recht B., Viwnyals O., *Understanding Deep Learning (Still) Requires Rethinking Generalization*, „Communications of the ACM” t. 64 nr 3 (2021), DOI: 10.1145/3446776.
246. Zhang F.-Y., Wang L.-L., Dong W.-W., Zhang M., Tash D., Li X.-J., Du S.-K., Yuan H.-M., Zhao R., Guan D.-W., *A preliminary study on early postmortem submersion interval (PMSI) estimation and cause-of-death discrimination based on nontargeted metabolomics and machine learning algorithms*, „International Journal of Legal Medicine” (2022), DOI: 10.1007/s00414-022-02783-4, https://doi.org/10.1007/s00414-022-02783-4.
247. Zhang N., Abraham A. (red.), *Proceedings of the Third International Symposium on Information Assurance and Security, IAS 2007, August 29-31, 2007, Manchester, United Kingdom*, IEEE Computer Society 2007.
248. Zhang Z., Geiger J., Pohjalainen J., Mousa A.E.-D., Jin W., Schuller B., *Deep Learning for Environmentally Robust Speech Recognition: An Overview of Recent Developments*, 21 września 2018 r., http://arxiv.org/abs/1705.10874.

249. Zhang Z., Suter D., Tian Y., Branzan Albu A., Sidère N., Jair Escalante H. (red.), *Pattern Recognition and Information Forensics: ICPR 2018 International Workshops, CVAUI, IWCF, and MIPPSNA, Beijing, China, August 20-24, 2018, Revised Selected Papers*, t. 11188, Cham 2019.
250. Zhao H., Zhou W., Chen D., Wei T., Zhang W., Yu N., *Multi-Attentional Deepfake Detection* [w:] 2021.
251. Zhong M., Tran K., Min Y., Wang C., Wang Z., Dinh C.-T., De Luna P., Yu Z., Rasouli A.S., Brodersen P., Sun S., Voznyy O., Tan C.-S., Askerka M., Che F., Liu M., Seifitokaldani A., Pang Y., Lo S.-C., Ip A., Ulissi Z., Sargent E.H., *Accelerated discovery of CO2 electrocatalysts using active machine learning*, „Nature" t. 581 nr 7807 (2020), DOI: 10.1038/s41586-020-2242-8.
252. Ziobroń A., *Deepfake a prawo karne. Uwagi „de lege lata" i „de lege ferenda" dotyczące fałszywej pornografii*, „Studenckie Prace Prawnicze, Administratywistyczne i Ekonomiczne" t. 37 (2021), DOI: 10.19195/1733-5779.37.15.

Wykaz równań

1.3.11. Pochodna funkcji kosztu wobec wag synaptycznych neuronu wyjściowego.

1.3.12. Reguła delta dla neuronu warstwy wyjściowej.

1.3.13. Pochodna funkcji kosztu wobec wag synaptycznych neuronu warstwy ukrytej.

1.3.14. Pochodna funkcji kosztu wobec sygnału z neuronu warstwy ukrytej.

1.3.15. Pochodna sygnału wobec pobudzenia neuronu warstwy ukrytej.

1.3.16. Pochodna pobudzenia wobec wag synaptycznych neuronu warstwy ukrytej.

1.3.17. Gradient lokalny błędu (delta) dla warstwy ukrytej.

1.3.18. Pochodna funkcji kosztu wobec wag synaptycznych neuronu warstwy ukrytej.

1.3.19. Reguła delta dla neuronu warstwy ukrytej.

1.3.20. Sztuczny neuron liniowy.

1.3.21. Reguła Hebba.

1.3.22. Funkcja sieci (*AlphaZero*).

1.3.23. Funkcja kosztu (*AlphaZero*).

1.4.1. Wymiary map aktywności na warstwach konwolucyjnych

1.4.2. Liczba parametrów warstwy konwolucyjnej.

1.4.3. Funkcja kosztu (wassersteinowska).

1.5.1. Wymiary map aktywności po zastosowaniu transponowanej konwolucji.

1.5.2. *Batch* sygnałów wyjściowych z danego neuronu.

1.5.3. Średnia sygnałów wyjściowych z danego neuronu.

1.5.4. Odchylenie standardowe sygnałów wyjściowych z danego neuronu.

1.5.5. Normalizacja sygnału wyjściowego z danego neuronu.

1.5.6. Korekta sygnału wyjściowego z danego neuronu.

5.0.1. *Modus tollendo tollens.*

5.0.2. *Modus tollendo tollens* dla hipotez statystycznych.

5.0.3. *Modus tollendo tollens* dla p-wartości.

5.0.4. Dowodzenie prawdziwości poprzednika przez prawdziwość następnika implikacji.

5.0.5. Dowodzenie prawdziwości hipotezy przez prawdziwość implikowanej hipotezy statystycznej.

8.1.1. Wassersteinowski wariant funkcji *hinge loss*.

Wykaz rysunków

Wykaz tabel