Autoreferat

1. Imię i nazwisko.

Łukasz Apiecionek

2. Posiadane dyplomy, stopnie naukowe lub artystyczne – z podaniem podmiotu nadającego stopień, roku ich uzyskania oraz tytułu rozprawy doktorskiej.

Doktor nauk technicznych w zakresie informatyki nadany uchwałą Rady Naukowej Instytutu Podstawowych Problemów Techniki Polskiej Akademii Nauk w dniu 27 stycznia 2011 roku. Tytuł pracy: Metoda oceny jakości transmisji głosowej w telefonii VoIP.

3. Informacja o dotychczasowym zatrudnieniu w jednostkach naukowych lub artystycznych.

01.10.2011 – 30.09.2019 – Uniwersytet Kazimierza Wielkiego w Bydgoszczy, adiunkt

01.10.2019 – Uniwersytet Kazimierza Wielkiego w Bydgoszczy, profesor uczelni badawczo-dydaktyczny, Wydział Informatyki, Katedra Systemów Inteligentnych

01.10.2024 – Uniwersytet Kazimierza Wielkiego w Bydgoszczy, profesor uczelni badawczo-dydaktyczny, kierownik Katedry Cyberbezpieczeństwa, Wydział Informatyki,

4. Omówienie osiągnięć, o których mowa w art. 219 ust. 1 pkt. 2 ustawy z dnia 20 lipca 2018 r. Prawo o szkolnictwie wyższym i nauce (Dz. U. z 2021 r. poz. 478 z późn. zm.). Omówienie to winno dotyczyć merytorycznego ujęcia przedmiotowych osiągnieć, jak i w sposób precyzyjny określać indywidualny wkład w ich powstanie, w przypadku, gdy dane osiągnięcie jest dziełem współautorskim, z uwzględnieniem możliwości wskazywania dorobku z okresu całej kariery zawodowej.

Określenie osiągnięcia naukowego będącego podstawą ubiegania się o nadanie stopnia doktora habilitowanego

Podstawą ubiegania się o nadanie stopnia są:

- implementacja liczb rozmytych w sztucznej sieci neuronowej przedstawiona w monografii,
- cykl publikacji dotyczących wykorzystania mechanizmów logiki rozmytej do zwiększenia poziomu bezpieczeństwa systemów i sieci teleinformatycznych,

- współautorstwo w oryginalnym rozwiązania technologicznym – Systemie Wspomagania Dowodzenia HMS C3IS JAŚMIN.

Pierwszym przedstawionym do oceny osiągnięciem jest opracowanie skalowanej rozmytej sztucznej sieci neuronowej wykorzystującej jedno z rozwiązań logiki rozmytej - skierowane liczby rozmyte. Szczegółowy zakres przedmiotowego osiągnięcia został przedstawiony w monografii: Łukasz Apiecionek, Liczby rozmyte w architekturze sieci neuronowych, Wydawnictwo UKW, 2024, ISBN 978-83-8018-657-6.

W swoim rozwiązaniu wykorzystałem skierowane liczby rozmyte. Liczby te zaproponował Prof. Witold Kosiński wraz ze swoim zespołem [1-2] poprzez wprowadzenie do nich trendu, co okazało się pomocne w wielu analizach zdarzeń. Skierowana liczba rozmyta (ang. Ordered Fuzzy Number) to uporządkowana para $A = (f_A, g_A)$ funkcji ciągłych $F : f_A, g_A : [0,1] \rightarrow R$ zwanych odpowiednio częściami:

- up_A – początkiem, zboczem narastającym:

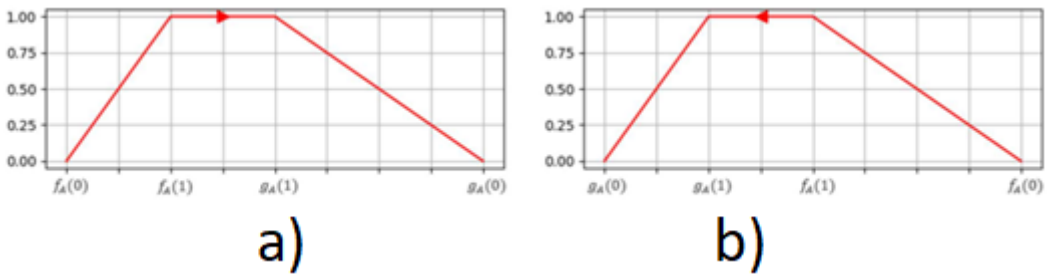
$up_A = \{(f_A(y), y) : y \in [0,1]\}$	1

- $down_A$ - końcem, zboczem opadającym:

$down_A = \{(g_A(y), y) : y \in [0,1]\}$	2

Kolejność wykresów funkcji f_A, g_A pozwala określić orientację skierowanej liczby rozmytej [3-4] (zobrazowane na rysunku 1):

- pozytywną w przypadku gdy skierowanie funkcji jest zgodne z narastaniem osi OX,
- negatywną w przypadku gdy skierowanie funkcji jest odwrotne do kierunku narastania osi OX.



Rys. 1. Przykładowa Skierowana Liczba Rozmyta a) pozytywna b) negatywna

Podstawowe operacje arytmetyczne na skierowanych liczbach rozmytych, mianowicie sumę, różnicę, mnożenie przez skalar, iloczyn oraz iloraz definiuje się następująco [5-7]:

- Suma: skierowana liczba rozmyta $C = (f_C, g_C)$ jest sumą liczb $A = (f_A, g_A)$ i $B = (f_B, g_B)$, gdy:

$\forall_{y\in[0,1]}[f_A(y)+f_B(y)=f_C(y)\ \wedge\ g_A(y)+g_B(y)=g_C(y)]$	3

- Różnica: skierowana liczba rozmyta $C=(f_C,g_C)$ jest różnicą liczb $A=(f_A,g_A)$ i $B=(f_B,g_B)$, gdy:

$\forall_{y\in[0,1]}[f_A(y)-f_B(y)=f_C(y)\ \wedge\ g_A(y)-g_B(y)=g_C(y)]$	4

- Mnożenie przez skalar: skierowana liczba rozmyta $C=(f_C,g_C)$ jest mnożeniem liczby $A=(f_A,g_A)$ przez skalar $r\in R$, jeżeli:

$\forall_{y\in[0,1]}[r\cdot f_A(y)=f_C(y)\ \wedge\ r\cdot g_A(y)=g_C(y)]$	5

- Iloczyn: Skierowana Liczba Rozmyta $C=(f_C,g_C)$ jest iloczynem liczb $A=(f_A,g_A)$ i $B=(f_B,g_B)$, jeżeli:

$\forall_{y\in[0,1]}[f_A(y)\cdot f_B(y)=f_C(y)\ \wedge\ g_A(y)\cdot g_B(y)=g_C(y)]$	6

- Iloraz: skierowana liczba rozmyta $C=(f_C,g_C)$ jest ilorazem liczb $A=(f_A,g_A)$ przez $B=(f_B,g_B)$, jeżeli:

$\forall_{y\in[0,1]}[f_A(y)/f_B(y)=f_C(y)\ \wedge\ g_A(y)/g_B(y)=g_C(y)]$	7

W celu powrotu z dziedziny skierowanych liczb rozmytych do dziedziny liczb rzeczywistych, np. w celu wysterowania urządzenia, gdzie konieczne jest podanie wartości rzeczywistej, przeprowadza się operację wyostrzania [8] . Oczywiście istnieją różne metody wyostrzania, a fakt, że uzyskuje się wówczas zupełnie różne wyniki wskazuje, że ich zastosowania mogą być różne. Naturalnie można podejrzewać, że w zależności od konkretnego zastosowania, wybór metody wyostrzania będzie pasował bardziej lub mniej. Inaczej mówiąc wyniki operacji wyostrzania będą optymalne dla jednego zastosowania, a zupełnie przeciwnie dla innego. W literaturze można znaleźć szczegółowe opisy wyostrzania, do najpopularniejszych metod należą: FOM, MOM, LOM, Golden ratio oraz Mandala. Metody FOM, MON I LOM wywodzą się ze zbiorów rozmytych, a ich definicje przedstawiają się następująco:

- FOM – First of Maxima, jest operatorem, który w wyniku zwraca najmniejszy element jądra zbioru A

$FOM(A)=\min\ core(A),$	8

- LOM – Last of Maxima, jest operatorem, którego wynikiem jest największy element jądra zbioru A

$LOM(A)=\max\ core(A),$	9

- MOM – Mean of Maxima jest operatorem, którego wynikiem jest średnia arytmetyczna sumy najmniejszego oraz największego elementu jądra zbioru A

$MOM(A)=\frac{\min core(A)+\max\ core(A)}{2}.$	10

Jak widać ze wzorów metody FOM, LOM i MOM nie uwzględniają skierowania, które posiadają skierowane liczby rozmyte. Uwzględniając definicje skierowanych liczb rozmytych,

m.in. fakt, że wartość funkcji f dla 0 wynosi $f(0)$ a dla 1 wynosi $f(1)$, oraz analogicznie dla funkcji g będą to wartości $g(0)$ i $g(1)$, wyprowadzono operatory wyostrzania:

$\phi FOM(f,g) = f(1),$	11
$\phi LOM(f,g) = g(1),$	12
$\phi MOM(f,g) = \frac{f(1)+g(1)}{2}.$	13

Kolejne dwie metody wyostrzania skierowanych liczb rozmytych to metoda Golden Ratio oraz Mandala:

- Golden ratio [9-10] zdefiniowany jako operator wyostrzania metodą złotego podziału umożliwia wyznaczenie wartości rzeczywistej z danej liczby rozmytej A w następujący sposób:

$GR(A) = \begin{cases} min(supp(A)) + \frac{\lvert supp(A)\rvert}{\phi} \ gdy\ skierowanie\ liczby\ A\ jest\ pozytywne \\ max(supp(A)) - \frac{\lvert supp(A)\rvert}{\phi} \ gdy\ skierowanie\ liczby\ A\ jest\ negatywne' \end{cases}$	14

gdzie ϕ jest właśnie współczynnikiem złotego podziału I wynosi $\phi = 1{,}618033998875$ ...

- Mandala [3] zdefiniowany jako operator wyostrzania został zainspirowany obrazami buddyjskich mnichów, jego definicja wygląda następująco:

$MF(A) = \begin{cases} f_A(0) + r_A \ gdy\ skierowanie\ liczby\ A\ jest\ pozytywne \\ g_A(0) - r_A \ gdy\ skierowanie\ liczby\ A\ jest\ negatywne' \end{cases}$	15

gdzie:

$r_A = \frac{1}{f_A(1)-f_A(0)} \int_{f_A(0)}^{f_A(1)} x dx - \frac{f_A(0)}{f_A(1)-f_A(0)} \int_{f_A(0)}^{f_A(1)} dx + \frac{g_A(0)}{g_A(0)-g_A(1)} \int_{g_A(1)}^{g_A(0)} dx - \frac{1}{g_A(0)-g_A(1)} \int_{g_A(1)}^{g_A(0)} x dx + \int_{f_A(1)}^{g_A(1)} dx$	16

oraz $r_A = 0$ dla skierowanej liczby rozmytej A, dla której funkcje f_A i g_A są stałe.

Wykorzystanie różnych metod wyostrzania spowoduje uzyskanie różnych wyników końcowych. W [11] autor pracy użył różnych metod wyostrzania do tego samego zadania. Zadanie polegało na użyciu istniejącego algorytmu zapobiegania atakom typu Denial of Services [12]. W przeprowadzonym eksperymencie, celem ataku był serwer. Założeniem było wykrycie ataku przez serwer, który najlepiej jest w stanie stwierdzić, że został właśnie celem zagrożenia, a następnie przekazanie informacji do routera/urządzenia brzegowego, aby rozpocząć proces blokowania specyficznego ruchu.

Podczas przeprowadzonego testu, sprawdzono wpływ różnych metod wyostrzania wyników: Golden ratio, Mandala, FOM, MOM, LOM. Żadna z użytych metod nie spowodowała

przeciążenia serwera, natomiast miały wpływ na zachowanie metody. W wyniku eksperymentu potwierdził się wpływ wyboru metody na docelowe rozwiązanie.

W przedstawionym przykładzie wykazano, że zastosowanie różnych metod wyostrzania będzie miało wpływ na końcowy wynik obliczeń, czy sterowania daną wielkością/zjawiskiem. Badanie to miało istotne znaczenie pod kątem użycia metody wyostrzania w późniejszym procesie budowy rozmytej sieci neuronowej. Otóż w momencie wyostrzenia wyniku i obliczenia błędu, w zależności od wybranej metody wyostrzania (Golden ratio, Mandala, FOM, MOM, LOM) uzyskiwane są różne wyniki. Efekt ten ma wpływ na proces uczenia sieci, zarówno w zakresie osiąganych wyników jak i szybkości samego procesu uczenia. Stąd zagadnienie wyostrzania zasługuje na szczególne podkreślenie istotności przy wykorzystaniu logiki rozmytej i skierowanych liczb rozmytych zarówno w sterowaniu procesami jak i sztucznych sieciach neuronowych.

W celu przygotowania rozmytej sztucznej sieci neuronowej wykorzystano neuron według pomysłu McCulloch-Pittsa, jednak wprowadzając, jak wspomniano, rozmycie wag. Arytmetyka zgodnie z teorią skierowanych liczb rozmytych odbywa się w następujących krokach:

Krok 1. Obliczenie sumy wejść dla poszczególnych neuronów według następującego wzoru:

$S = W_0 + \sum_{i=1}^{n} X_i W_i$	17

Krok 2. Obliczenie aktywacji wyjść neuronów według następującego wzoru:

$Y = f(S)$	18

Gdzie przedstawione we wzorach parametry S, W, X, Y, są liczbami rozmytymi w notacji skierowanych liczb rozmytych. Oczywiście takie podejście do budowy sieci wymagało rozwiązania między innymi następujących problemów:

- opracowania sposobu rozmywania danych wejściowych do sieci,
- opracowania sposobu wyostrzania wyników wyjściowych uzyskiwanych w sieci,
- opracowania algorytmów uczenia sieci, które będą operowały w dziedzinie liczb rozmytych.

Pierwsza implementacja sieci rozmytej z zastosowaniem rozmytych sieci neuronowych działała wykorzystując na wejściu dane rozmyte i zwracała wynik w postaci liczby rozmytej [13]. Implementacja ta obarczona była problemem wprowadzenia do danych tylko w postaci singletonów. Dlatego w przeprowadzonych badaniach z wykorzystaniem znanego zbioru danych przedstawiających rodzaje kwiatu Irysa [14] zbiór danych trzeba było ograniczyć do

dwóch klas. Dopiero wtedy możliwe było porównanie sieci z siecią głęboką. Na potrzeby testu zbiór został ograniczony do dwóch klas: Setosa, Veriscolour.

Podczas badania wykorzystano następujące architektury sieci:

- Sieć głęboka 1: warstwa wejściowa – 4 neurony, warstwa głęboka – 8 neuronów, warstwa wyjściowa – 2 neurony,
- Rozmyta sieć neuronowa 1: warstwa wejściowa – 4 neurony, warstwa głęboka – 1 neuron, warstwa wyjściowa – 2 neurony,
- Rozmyta sieć neuronowa 2: warstwa wejściowa – 4 neurony, warstwa głęboka – 3 neurony, warstwa wyjściowa – 2 neurony,
- Rozmyta sieć neuronowa 3: warstwa wejściowa – 4 neurony, warstwa głęboka – 4 neurony, warstwa wyjściowa – 2 neurony,
- Rozmyta sieć neuronowa 4: warstwa wejściowa – 4 neurony, warstwa głęboka – 8 neuronów, warstwa wyjściowa – 2 neurony.

Podczas uczenia sieci, dla tradycyjnej sieci głębokiej wystarczyło 500 epok, natomiast dla sieci rozmytych używano 550 epok. Ideą testu było rozpoznanie kwiatu, co oznacza, że wynikiem było 0 lub 1 odnośnie przynależności do danej klasy/gatunku. Dlatego jako wynik, zebrano osiągnięte parametry w zakresie rozpoznawania kwiatu i obliczono liczbę prawidłowo rozpoznanych kwiatów. Wynik procentowy błędnie rozpoznanych kwiatów zebrano w tabeli 1.

Tabela 1. Zestawienie wyników osiągniętych przez badane sieci do rozpoznawania kwiatu Irysa

Lp.	Sieć	Procent błędnie rozpoznanych kwiatów [%]
1	Sieć głęboka 1	4,22
2	Rozmyta sieć neuronowa 1	0,03
3	Rozmyta sieć neuronowa 2	0,61
4	Rozmyta sieć neuronowa 3	0,28
5	Rozmyta sieć neuronowa 4	0,61

Jak można zauważyć w tabeli 1, osiągnięto bardzo wysoką dokładność działania sieci.

Warto zauważyć, że bardzo dobre wyniki działania uzyskano również dla sieci z mniejszą liczbą neuronów w warstwach ukrytych, Takie założenie przyświecało autorowi rozwiązania. Założono mianowicie, że fakt rozmycia wag odpowiadających za aktywację neuronów pozwoli na rozwiązywanie tych samych zagadnień jak sieci głębokie, ale z mniejszą liczbą warstw i mniejszą liczbą neuronów. Jednocześnie zastosowanie prostej arytmetyki skierowanych liczb rozmytych, nie spowoduje wydłużenia czasu uzyskiwania odpowiedzi sieci, czy procesu uczenia. Dzięki temu, takie rozmyte sieci powinny znaleźć zastosowanie we wszystkich miejscach, gdzie nie ma dostępu do dużej mocy obliczeniowej, np. w urządzeniach końcowych, czy w urządzeniach Internetu Rzeczy. Dalej, pozwoli to na prowadzenie procesu analizy danych

w miejscach ich powstawania, bez konieczności przesyłania danych do chmury, czy centrum obliczeniowego. Brak konieczności przesyłania danych do jednego centrum analiz pozwoli na:

- zmniejszenie ruchu sieciowego – wyłączenie konieczności transmisji danych,
- zmniejszenie zapotrzebowania na energię potrzebną do transmisji danych, co jest bardzo ważne np. w przypadku systemów zasilanych z akumulatorów czy systemów transmitujących dane drogą radiową.

Jak już wspomniano, implementacja ta miała ograniczenie w postaci wprowadzenia danych tylko w postaci zbiorów jednoelementowych. Ponadto, ponieważ wszystkie obliczenia odbywały się w dziedzinie liczb rozmytych, zgodnie z arytmetyką skierowanych liczb rozmytych, nie następowało „wiązanie" składników liczb rozmytych. Oznacza to, że pracując z taką siecią, można ją zasymulować jako cztery równolegle działające sieci głębokie.

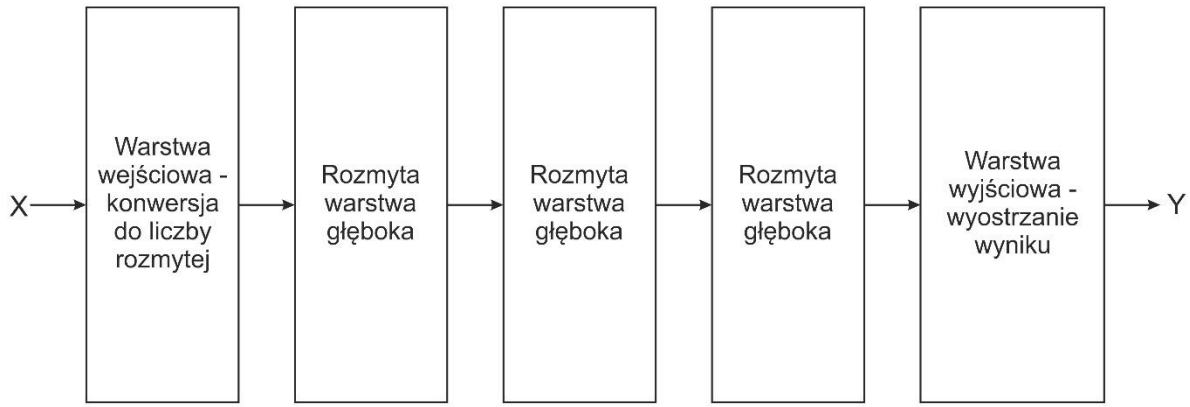
Stąd powstałą idea implementacji sieci rozmytej w pełni skalowanej, z zastosowaniem wyostrzania wyniku do dziedziny liczb rzeczywistych, co dopiero pozwala na połączenie składników zastosowanych skierowanych liczb rozmytych i skorzystanie z ich cech, a mianowicie wskazywania trendów. Założono, że takie podejście według postawionej hipotezy powinno pozwolić na uzyskiwanie wyników działania sieci jak sieci głębokie ale przy mniejszych architekturach oraz wzrost szybkości uczenia sieci.

Dlatego opracowano rozwiązanie, którego schemat przedstawia rysunek 2. W tym modelu warstwowym wyszczególniono:

- warstwę wejściową służącą do konwersji danych wejściowych do dziedziny liczb rozmytych, czyli rozmywanie danych wejściowych,
- ukryte warstwy głębokie,
- warstwę wyjściową sieci, w której odbywa się proces wyostrzania wyniku.

Oczywistym jest, ze proces rozmywania danych jest procesem zależnym od ich charakteru, nie można od tak stwierdzić, że w warstwie tej po prostu on zachodzi. Proces ten wymaga analizy charakteru danych i będzie się różnił w zależności od dziedziny i rodzaju zadania, które ma rozwiązań opracowywana sieć. Na schemacie sieci przedstawiono trzy warstwy głębokie, aczkolwiek ideą opracowywania sieci i przygotowywania jej kodu, było opracowanie sieci uniwersalnej, w której będzie możliwe zdefiniowanie dowolnej liczby warstw ukrytych (również co do liczby neuronów w warstwach). Dlatego sieć tą należy traktować właśnie jako uniwersalną. Natomiast proces wyostrzania, jak pokazano przy omawianiu zagadnienia skierowanych liczb rozmytych, również będzie miał wpływ na uzyskiwane wyniki. Ponieważ

jednak, w tym zakresie, opracowano dość duży zbiór metod wyostrzania, nie było konieczności opracowywania nowych metod w celu przygotowania algorytmów uczenia.



Rys. 2. Ogólna architektura zaproponowanej sieci rozmytej

Podczas prac badawczych zaszła konieczność opracowania algorytmów uczenia sieci dostosowanych do implementacji sieci. W ramach implementacji sieci zdefiniowano klasę OFN (skierowanych liczb rozmytych), która później byłą wykorzystywana do przechowywana jako obiekt macierzy NumPy. Następnie w celu wykonywania operacji arytmetycznych na obiekcie klasy skierowanej liczby rozmytej, zdefiniowano operatory dodawania, odejmowania, mnożenia i dzielenia. Ponadto zdefiniowano operatory rzutowania klasy na inne obiekty, co pozwala na używanie jej w dowolnych obliczeniach, dobierając odpowiedni sposób rzutowania. Opracowaną sieć porównano w działaniu z siecią głęboką zbudowaną za pomocą biblioteki Tensorflow. W pierwszej kolejności zbadano realizację przez obie sieci funkcji liniowej oraz operacji logicznych OR, AND i XOR. W dalszej części opisu osiągnięcia przedstawiono wybrane wyniki badań.

Celem badań nie było poszukiwanie optymalnej architektury, tylko sprawdzenie możliwości sieci w porównaniu z tradycyjną siecią głęboką oraz weryfikacja algorytmu i procesu uczenia. Analogiczne prace przeprowadzono dla sieci głębokiej. Zdefiniowano architekturę sieci o takich samych parametrach jak sieć rozmyta. Do utworzenia sieci wykorzystano interfejs Keras do biblioteki Tensorflow [15].

Uzyskane wyniki wykazały poprawność implementacji rozwiązania. Zaletą jednak zaproponowanej sieci rozmytej, w stosunku do sieci głębokiej zbudowane z wykorzystaniem biblioteki Tensorflow i zakodowanego w niej algorytmu uczenia, jest to, że proces uczenia trwa znacznie krócej. Porównanie czasów uczenia wypada bardzo korzystnie na rzecz sieci rozmytej, której zakodowany algorytm potrafi zakończyć proces uczenia w 9 sekund, podczas gdy proces

uczenia sieci głębokiej trwał 7 minut i 22 sekundy. Oczywiście porównanie czasów dotyczy wykonania kodu na tym samym środowisku programowo-sprzętowym. Zakładając użycie takiej samej maszyny, proces uczenia sieci rozmytej był nawet 49 razy szybszy. Wyniki działania sieci przedstawiono w tabelach.

Tabela 2. Zestawienie wyników osiągniętych przez badane sieci – funkcja liniowa, liczba epok 10000

Wartość argumentu x	Rozmyta sieć neuronowa	Sieć głęboka	Rozmyta sieć neuronowa	Sieć głęboka
	[y=2*x]	[y=2*x]	Błąd względny [%]	Błąd względny [%]
Zbiór uczący				
0	0,00833068	0,0000000167638060	0,83	0,00
0,1	0,18690828	0,19999999	6,55	0,00
0,2	0,4006355	0,4	0,16	0,00
0,3	0,61514937	0,6	2,52	0,00
0,4	0,78635874	0,8	1,71	0,00
Zbiór testowy				
0,05	0,09208328	0,10000001	7,92	0,00
0,15	0,2909572	0,30000004	3,01	0,00
0,25	0,51072344	0,5	2,14	0,00
0,35	0,70830254	0,7	1,19	0,00
0,45	0,84797919	0,9	5,78	0,00
Wartość średnia:			3,18	0,00

Tabela 3. Zestawienie wyników osiągniętych przez badane sieci – funkcja logiczna OR, liczba epok 10000

Wartość argumentów wejściowych	Wartość wyjściowa	Rozmyta sieć neuronowa	Sieć głęboka
Zbiór uczący			
0 0	0	0,0000317	0,25
0 1	1	0,9962043	0,75
1 0	1	0,99619815	0,75
1 1	1	0,99932649	1,25
Wynik działania sieci po zaokrągleniu wartości:			
0 0	0	0	0

0 1	1	1	1
1 0	1	1	1
1 1	1	1	1

Tabela 4. Zestawienie wyników osiągniętych przez badane sieci – funkcja logiczna AND, liczba epok 10000

Wartość argumentów wejściowych	Wartość wyjściowa	Rozmyta sieć neuronowa	Sieć głęboka
Zbiór uczący			
0 0	0	-0,0304225	-0,25000006
0 1	0	-0,0010326	0,24999996
1 0	0	0,00581458	0,25
1 1	1	0,98292154	0,7499999
Wynik działania sieci po zaokrągleniu wartości:			
0 0	0	0	0
0 1	0	0	0
1 0	0	0	0
1 1	1	1	1

Przedstawione rozwiązanie rozmytej sieci neuronowej jest rozwiązaniem niespotykane dotychczas w publikacjach naukowych, poprzez propozycję budowy sieci rozmytej analogiczną do sieci głębokich, dzięki czemu uzyskujemy wszystkie zalety sieci głębokich przy jednoczesnym zastosowaniu zalet logiki rozmytej. Prosta arytmetyka pozwala na szybkie obliczenia przy jednoczesnym uzyskaniu możliwości wnioskowania i generalizacji wiedzy. Ponadto prostota obliczeń pozwoliła na uzyskanie bardzo szybkiego algorytmu uczenia. Wzrost szybkości uczenia w stosunku do procesu uczenia sieci głębokiej z użyciem biblioteki Tensorflow jest znaczny i wynosił o 8 do 50 razy. Po drugie jest to nowe podejście pod względem inżynierskim. Zakodowanie arytmetyki skierowanych liczb rozmytych w postaci klasy wraz z przeciążeniem operatorów arytmetycznych pozwala na użycie jej z powszechnie stosowaną przez badaczy biblioteką NumPy. Takie rozwiązanie umożliwia prowadzenie badań z użyciem rozmytej sieci neuronowej dla dowolnego zagadnienia. Jest to otwarcie drzwi do dalszych prac i analiz zastosowania rozmytych sieci neuronowych w połączeniu ze skierowanymi liczbami rozmytymi dla dowolnych zagadnień i zastosowań.

Podsumowując, w celu przygotowania sieci wykorzystano neuron według pomysłu McCulloch-Pittsa, jednak wprowadzając, jak wspomniano, rozmycie wag.

Oczywiście wielkości używane w sieci jak wektory wejściowe, wyjściowe, wagi neuronów są wartościami rozmytymi w notacji skierowanych liczb rozmytych. Takie podejście do budowy sieci wymagało rozwiązania szeregu problemów implementacyjnych. Problemy te dotyczyły takich aspektów działania i procesu uczenia sieci jak:

- opracowania sposobu rozmywania danych wejściowych do sieci, które w tradycyjnej formie mają charakter liczb rzeczywistych, a konieczne było przejście do dziedziny liczb rozmytych,
- opracowania/dobrania sposobu wyostrzania wyników wyjściowych uzyskiwanych w sieci w związku z faktem, że celem budowy sieci było zastąpienie sieci głębokich przez rozmyte sieci neuronowe, więc wynik musiał również zostać poddany wyostrzaniu,
- opracowania algorytmów uczenia sieci, które będą operowały w dziedzinie liczb rozmytych, gdyż dotychczasowe algorytmy nie operowały w tej dziedzinie.

Zaproponowano więc warstwowy model rozmytej sieci neuronowej z arytmetyką skierowanych liczb rozmytych. W tym modelu warstwowym wyszczególniono:

- warstwę wejściową służącą do konwersji danych wejściowych do dziedziny liczb rozmytych, czyli rozmywanie danych wejściowych,
- ukryte warstwy głębokie, z możliwością definicji dowolnej liczby warstw i dowolnej liczby neuronów, jak w sieciach głębokich,
- warstwę wyjściową sieci, w której odbywa się proces wyostrzania wyniku.

Od momentu podjęcia badań nad opracowywaną siecią, oczywistym faktem było, ze proces rozmywania danych będzie procesem zależnym od charakteru danych. Założono, że nie będzie można od tak zdefiniować uniwersalnej metody rozmywania danych, a proces ten będzie wymagał za każdym razem analizy charakteru danych i będzie się różnił w zależności od dziedziny i rodzaju zadania, które ma rozwiązań opracowywana sieć. Dlatego w ramach definicji sieci proces rozmywania oznaczono warstwą rozmywania danych, której zadaniem jest konwersja danych do dziedziny liczb rozmytych. Dokładny proces rozmywania jest zagadnieniem bardzo ciekawym, często pomijanym w publikacjach, a jest to zagadnienie interesujące i wymagające dalszego zgłębiania w celu umożliwienia szerokiego zastosowania systemów rozmytych.

Założeniem projektowanej sieci było odzwierciedlenie architektury sieci głębokich, dlatego w zaproponowanym modelu można wprowadzać dowolną liczbę warstw ukrytych sieci

z dowolną liczbą neuronów w definiowanych warstwach. Takie podejście pozwoliło traktować ją jako sieć uniwersalną.

Proces wyostrzania wyników ma wpływ na uzyskiwane wyniki. Jednakże w literaturze można już znaleźć bardzo duży zbiór metod wyostrzania i proces ten nie stworzył żadnych trudności podczas opracowywania sieci i jej algorytmu uczenia.

W celu łatwego używania zaproponowanej rozmytej sieci neuronowej z arytmetyką skierowanych liczb rozmytych przygotowano kod sieci w języku Python z wykorzystaniem szeroko używanych standardowych bibliotek do prac nad zagadnieniami Sztucznej Inteligencji. Zaproponowana budowa sieci rozmytej, poprzez dodanie arytmetyki skierowanych liczb rozmytych i zakodowanie jej w postaci klasy, którą można używać z biblioteką NumPy jest rozwiązaniem uniwersalnym, możliwym do przygotowania dowolnej architektury sieci. Prosta arytmetyka pozwala na szybkie obliczenia przy jednoczesnym uzyskaniu możliwości wnioskowania i generalizacji wiedzy. Ponadto prostota obliczeń pozwoliła na uzyskanie bardzo szybkiego algorytmu uczenia. Wzrost szybkości uczenia w stosunku do procesu uczenia sieci głębokiej z użyciem biblioteki Tensorflow jest znaczny i wynosił o 8 do 50 razy. Potwierdziły się inne badania literaturowe, które wskazywały na przyspieszenie procesu uczenia przy zastosowaniu logiki rozmytej.

Bibliografia:

[1] Kosiński W., Prokopowicz P., Ślęzak D., On Algebraic Oprerations on Fuzzy Numbers, Inteligent Information Processing and Web Mining: proceedings of the International IIS:IIPWM'03 Conference held In Zakopane , Poland, June 2-5, 2003

[2] Kosiński W., On Fuzzy Number Calculus, Int. J. Appl. Math. Comput.Sci., Vol. 16, No. 1, 2006, pp.51-57

[3] Prokopowicz P., Czerniak J., Mikołajewski D., Apiecionek Ł, Ślęzak D., Theory and applications of ordered fuzzy numbers: a tribute to Professor Witold Kosiński, Springer Nature, 2017

[4] Czerniak J.M., Dobrosielski W.T., Apiecionek Ł., Ewald D., Paprzycki M., Practical Application of OFN Arithmetics in a Crisis Control Center Monitoring. In: Fidanova S., Recent Advances in Computational Optimization. Studies in Computational Intelligence, Vol 655, Springer, 2016, s.53-56

[5] Kosiński W., Prokopowicz P., Mathematica Applicanda, Vol 32, No 46/05, Polskie Towarzystwo Matematyczne, 2004

[6] Kosiński W., Prokopowicz P., Ślęzak D., Ordered Fuzzy Numbers Bulletin of the Polish Academy of Sciences. Mathematics. Vol 51, No 3, Instytut Matematyczny PAN, 2003
[7] Czerniak, J.M., Dobrosielski, W., Zarzycki, H., Apiecionek, Ł. (2015). A Proposal of the New owlANT Method for Determining the Distance between Terms in Ontology. In: Filev, D., et al. Intelligent Systems'2014. Advances in Intelligent Systems and Computing, vol 323. Springer, Cham. https://doi.org/10.1007/978-3-319-11310-4_21
[8] Bednarek T., Kosiński W., Węgrzyn-Wolska K., On Orientation Sensitive Defuzzification Functionals. In: Rutkowski, L., Korytkowski, M., Scherer, R., Tadeusiewicz, R., Zadeh, L.A., Zurada, J.M. (eds) Artificial Intelligence and Soft Computing. ICAISC 2014. Lecture Notes in Computer Science(), Vol 8468, Springer, 2014, s.658
[9] Dobrosielski W., Czerniak J., Szczepański J., Zarzycki H., Two new defuzzification methods useful for different fuzzy arithmetics. In International Workshop on Intuitonistic Fuzzy Sets and Generalized Nets (pp. 83-101), Springer, 2016, s.90-93
[10] Dobrosielski W., Szczepański J., Zarzycki H., A proposal for a Method of Deffuzification Based on the Golden Ratio – GR. In: Novel Developments in Uncertainty Representation and Processing. Advances in Intelligent Systems and Computing, Vol 401, Springer, 2016, s.80-82
[11] Apiecionek, Ł. (2022). Defuzzification Method Comparison in Real Data Packets Control Method for Network Protection. In: Choraś, M., Choraś, R.S., Kurzyński, M., Trajdos, P., Pejaś, J., Hyla, T. (eds) Progress in Image Processing, Pattern Recognition and Communication Systems. CORES IP&C ACS 2021 2021 2021. Lecture Notes in Networks and Systems, vol 255. Springer, Cham. https://doi.org/10.1007/978-3-030-81523-3_27
[12] Apiecionek Ł., Makowski W., Intelligent FTBint Method for Server Resources Protec-tion, Beyond Databases, Architectures and Structures. Advanced Technologies for Data Mining and Knowledge Discovery, Volume 613 of the series Communications in Com-puter and Information Science pp 683-691, 2016
[13] Apiecionek, Ł.; Moś, R.; Ewald, D. Fuzzy Neural Network with Ordered Fuzzy Numbers for Life Quality Technologies. Appl. Sci. 2023, 13, 3487. https://doi.org/10.3390/app13063487
[14] http://archive.ics.uci.edu/ml/datasets/Iris, dostęp online 04.11.2022
[15] https://keras.io/, dostęp online 09.08.2023

Drugim przedstawionym do oceny osiągnięciem jest cykl publikacji dotyczących wykorzystania mechanizmów logiki rozmytej do zwiększenia poziomu bezpieczeństwa systemów i sieci teleinformatycznych. Cykl stanowi zbiór ośmiu publikacji:

[C1] Łukasz Apiecionek, Wojciech Makowski, Intelligent FTBint Method for Server Resources Protection, W: Beyond Databases, Architectures and Structures. Advanced Technologies for Data Mining and Knowledge Discovery, 12th International Conference, BDAS 2016, Ustroń, Poland, May 31 - June 3, 2016, Proceedings / Ed. Stanisław Kozielski, Dariusz Mrozek, Paweł Kasprowski, Bożena Małysiak-Mrozek, Daniel Kostrzewa, Cham : Springer International Publishing, 2016, Communications in Computer and Information Science, 1865-0929 ; vol. 613, Strony: 683-691, ISBN: 978-3-319-34098-2, DOI: 10.1007/978-3-319-34099-9_53

[C2] Łukasz Apiecionek, Fuzzy Observation of DDoS Attack, W: Theory and Applications of Ordered Fuzzy Numbers. A Tribute to Professor Witold Kosiński / ed. Piotr Prokopowicz, Jacek Czerniak, Dariusz Mikołajewski, Łukasz Apiecionek, Dominik Ślęzak.
Cham : Springer International Publishing, 2017, Studies in Fuzziness and Soft Computing, 1434-9922; 356, Strony: 239-252, ISBN: 978-3-319-59613-6, DOI: 10.1007/978-3-319-59614-3_1442/76

[C3] Łukasz Apiecionek, Fuzzy Control for Secure TCP Transfer, W: Theory and Applications of Ordered Fuzzy Numbers. A Tribute to Professor Witold Kosiński / ed. Piotr Prokopowicz, Jacek Czerniak, Dariusz Mikołajewski, Łukasz Apiecionek, Dominik Ślęzak, Cham : Springer International Publishing, 2017, Studies in Fuzziness and Soft Computing, 1434-9922 ; 356, Strony: 253-268, ISBN: 978-3-319-59613-6, DOI: 10.1007/978-3-319-59614-3_15

[C4] Łukasz Apiecionek, Marcel Grossmann, Udo Krieger, Tytuł: Harmonizing IoT-Architectures with Advanced SecurityFeatures - A Survey and Case Study, Journal of Universal Computer Science - 2019, Vol. 25, no 6, pp. 571-590, p-ISSN: 0948-695x e-ISSN: 0948-6968, 2019, DOI: 10.3217/jucs-025-06-0571

[C5] Łukasz Apiecionek, Mateusz Biedziak, Fuzzy Adaptive Data Packets Control Algorithm for IoT System Protection, Journal of Universal Computer Science - 2020, Vol. 26, iss. 11, pp. 1435-1454, p-ISSN: 0948-695x e-ISSN: 0948-6968, 2020, DOI: 10.3897/jucs.2020.076

[C6] Apiecionek, Ł. (2022). Defuzzification method comparison in real data packets control method for network protection. In M. Choraś, R. S. Choraś, M. Kurzyński, P. Trajdos, J. Pejaś, & T. Hyla (Eds.), Progress in image processing, pattern recognition and communication systems: Proceedings of the conference (CORES, IP&C, ACS) - June 28-30 2021 (Vol. 255, pp. 276–283). Cham: Springer. https://doi.org/10.1007/978-3-030-81523-3_27

[C7] Łukasz Apiecionek, Fully Scalable Fuzzy Neural Network for Data Processing, Sensors - 2024, Vol. 24, iss. 16, art. no 5169., p-ISSN: 1424-8220, DOI: 10.3390/s24165169

[C8] Apiecionek, Ł., & Karbowski, P. (2024). Fuzzy Neural Network for Detecting Anomalies in Blockchain Transactions. Electronics, 13(23), 4646. https://doi.org/10.3390/electronics13234646

W 2011 roku, gdy autor wniosku rozpoczynał pracę na stanowisku adiunkta, rozwiązaniem problemów ataków typu Denial of Service było zalecenie aby zwiększać zasoby serwera. Stąd powstało zainteresowanie opracowaniem rozwiązania, które będzie potrafiło wykryć a następnie zablokować niepożądany ruch do serwera, co pozwoli zapobiec jego przeciążeniu. W ramach badań powstało *rozwiązanie inteligentnego mechanizmu usuwania pakietów z kolejki bramy dostępowej (firewalla) do sieci wewnętrznej z chronionym serwerem*, przedstawione w publikacji [C1, wnioskodawca był autorem hipotezy, określił proces badawczy, przygotował kod metody, przeprowadził badania, opracował wyniki, zweryfikował hipotezę oraz przygotował manuskrypt]. Zadaniem zaprojektowanego mechanizmu, jest kontrola ruchu przychodzącego na brzegu sieci przez urządzenie brzegowe, które dopuszcza ruch dozwolony, a blokuje ruchu zabroniony. W momencie ataku, mechanizm ma za zadanie blokować ruch na otwartym porcie do serwera. W przypadku, gdy liczba pakietów kierowanych do serwera przekroczy ustalony limit (packet_limit) w oknie czasowym (t), mechanizm przełącza się w tryb działania w fazie ataku. W tym momencie, z serwera pobierana jest lista adresów IP, które prowadziły poprawną komunikację z serwerem przed atakiem. Pakiety z tych adresów IP nie są usuwane z kolejki. Natomiast liczba nowych połączeń jest ograniczana w zdefiniowanym oknie czasowym (t) do ustalonej dozwolonej liczby (packet_counter). W przypadku, gdy liczba przekroczeń dozwolonego limitu powtarza się przez kolejne okna czasowe, wartość dozwolonej liczby pakietów jest zmniejszana. Dzieje się tak dlatego, że sieć zmaga się wówczas ze wzmożonym atakiem na serwer. W momencie, gdy atak ustaje, dozwoloną liczbę pakietów na dane okno czasowe zwiększa się stopniowo obserwując działanie sieci. Takie działanie mechanizmu zostało zweryfikowane eksperymentalnie poprzez symulację ataku na serwer. Manipulacja liczbą pakietów przekierowywanych do serwera pozwala na ciągłą obsługę prawidłowych użytkowników serwera. Pseudokod działania mechanizmy jest następujący:

```
packet_counter:=packet_counter + 1
if packet_counter < packet_limit then
 packet pass
else
 begin
```

```
 if IP address in listIP then
  packet pass;
 else
  packet drop;
end;
if times_slots ends then
begin
 if packet_counter>packet_limit then
  overdrop_times=overdrop_times + 1;
 packet_counter=0;
 if overdrop_times> subsequent_time_limit then
  packet_limit=packet_limit/2;
  overdrop_times=0;
 else
  packet_limit=packet_limit*2;
  overdrop_times=0;
end;
```

Wartość packet_counter zawiera liczbą pakietów, która może być przesłana w danym oknie czasowym. Jeśli wartość nie przekracza ustalonego limitu packet_limit, pakiet jest przesyłany, w przypadku gdy limit zostanie przekroczony, sprawdzane są kolejne warunki. Jeśli pakiet pochodzi z listy adresów dozwolonych, poprawnych połączeń przed atakiem na serwer, przechowywanych w zmiennej listIP, pakiet jest przesyłany, w przeciwnym przypadku jest usuwany z kolejki do przesłania (drop). Jeśli upłynie ustawiony limit czasu time_slots, następuje weryfikacja, czy przekroczy dozwoloną liczbę pakietów i następuje korekta dozwolonego limitu.

Wynik implementacji został sprawdzony eksperymentalnie poprzez wykonanie przykładowego ataku i obserwacje działania serwera. Zaimplementowanie metody pozwoliło serwerowi na dalszą pracę mimo ataku na jego zasoby [C1]. Do zalet przedstawionego rozwiązania należą m.in.:

- utrzymanie maksymalnego zużycia pamięci RAM przez serwer na ustalonym poziomie,
- umożliwienie serwerowi na dalszą pracę mimo ataku,
- umożliwienie użytkownikom sprzed ataku na dalszą pracę,
- brak konieczności działań po stronie administratora w momencie ataku,

- przywracanie stanu pierwotnego działania sieci po ataku.

Wynik analiz działania algorytmu skłonił autora do refleksji, w jaki sposób rozpoznać właściwy moment ataku na zasoby serwera. Licznik pakietów może nie być prawidłowym rozwiązaniem, gdyż w momencie wzmożonego ruchu prawidłowych użytkowników, może nastąpić nieprawidłowe wnioskowanie o rozpoznaniu ataku. Stąd została opracowana *autorska metoda rozpoznawania ataków za pomocą logiki rozmytej, a dokładnie z użyciem skierowanych liczb rozmytych*, przedstawiona w publikacji [C2]. Obserwując liczniki pakietów na poszczególnych routerach można zbudować skierowaną liczbę rozmytą, za pomocą której analizując jej skierowanie podjąć decyzję, czy nastąpił moment ataku na serwer. W celu dokonania obliczeń zaproponowano algorytm działania rozwiązania operujący na:

- zbudowaniu bezpiecznego tunelu pomiędzy urządzeniami w sieci (routerami, firewallami),
- przesyłania statystyk działania urządzeń do serwera za pomocą bezpiecznego tunelu i protokołu SNMP,
- dokonania obliczeń za pomocą skierowanych liczb rozmytych i podjęcie decyzji o ataku na serwer,
- przekazania do urządzeń brzegowych instrukcji o konieczności blokowania ruchu atakującego.

Wykrywanie ataku następuje poprzez zliczanie statystyk z urządzeń brzegowych, których wartości w poszczególnych slotach czasowych dostarczają składników do określenia skierowanej liczby rozmytej. Następnie określono definicje rozmytej obserwacji routera i grupy routerów, dzięki którym możliwe jest podjęcie decyzji o wykryciu ataku na sieć.

To rozwiązanie również zostało zweryfikowane eksperymentalnie poprzez symulację ataku na serwer z kilku routerów. Wyniki potwierdziły poprawność zdefiniowanej metody wykrywania ataku.

Idę rozmytej obserwacji działania sieci wykorzystano również do opracowania *autorskiego mechanizmu zwiększenia bezpieczeństwa transmisji danych z wykorzystaniem protokołu Multi-Path TCP*, przedstawioną w publikacji [C3]. W zaproponowanym rozwiązaniu dane są dzielone na bloki, które następnie są ustawiane w losowej kolejności, a następnie po zaszyfrowaniu przesyłane za pomocą różnych interfejsów dostępowych zgodnie z zasadą działania Multi-Path TCP. Problemem w takim przesyłaniu danych może być występowanie dużej liczby błędów w jednym z wybranych kanałów transmisji. Wówczas konieczne jest dokonywanie retransmisji, co zwiększa oczywiście czas niezbędny do przesłania całości danych. Dlatego

zaproponowano metodę wykrywania błędów w torze transmisyjnym z użyciem skierowanych liczb rozmytych. W kolejnych slotach czasowych zbierane są informację o liczbie wystąpień błędów. Następnie cztery kolejne wartości pozwalają na zbudowanie skierowanej liczby rozmytej. Podobnie jak w obserwacji ataków typu DoS, określono definicję rozmytej obserwacji kanału oraz możliwości podjęcia decyzji o zaprzestaniu używania tegoż kanału transmisji w momencie zauważenia wzmożonej liczby błędów wymagających retransmisji danych. Na tej podstawie określono algorytm obliczeń błędów w torach transmisyjnych oraz możliwość definiowania progów zaprzestania używania danego kanału. Dzięki temu możliwe jest zaoszczędzenie czasu i energii niezbędnej do retransmisji danych. Rozwiązanie zostało zweryfikowane podczas symulacji transmisji danych w kanałach z symulowaną stopą błędów, co wykazało poprawność implementacji i zysk dla transmitującego.

W ramach prac nad bezpieczeństwem rozwiązań *przygotowano również propozycję metodologii projektowania architektury rozwiązań IoT*, przedstawioną w publikacji [C4, wnioskodawca był współautorem hipotezy i procesu badawczy, przygotował środowisko i przeprowadził badania, współpracował przy opracowaniu wyników oraz weryfikacji hipotezy, a następnie przygotował manuskrypt]. Ideą, która przyświecała badaniu było znalezienie uniwersalnej architektury dla rozwiązań IoT z uwzględnieniem aspektów bezpieczeństwa systemów [C4]. W wyniku badań ustalono, że wpływ na wybór docelowej architektury ma wiele czynników związanych z wymogami funkcjonalnymi rozwiązania. Badania wykazały zasadność budowy modeli warstwowych i wybór docelowego rozwiązania na bazie m.in. wymagań systemowych, aspektów użytkowych, bezpieczeństwa, rodzaju i typu danych. Przygotowana publikacja jest przeglądem i przewodnikiem dla administratorów i projektantów rozwiązań IoT. Ciekawym wynikiem badań jest fakt, że nawet rozwiązania IoT budowane na urządzeniach typu Raspberry Pi potrafią zestawiać tunele IPSec z dość dużą wydajnością. W ramach zwiększenia bezpieczeństwa rozwiązań, w urządzenia IoT również można zastosować rozmytą obserwację potencjalnych ataków. Badania te były prowadzone w ramach projektu Miniatura 1 numer 2017/01/X/ST6/00613 finansowane przez Narodowe Centrum Nauki.

Rozmytą obserwację obciążenia zasobów zastosowano również do opracowania *autorskiej metody inteligentnej regulacji w mechanizmie ograniczania pakietów podczas ataku DoS na zasoby serwera*, przedstawioną w publikacji [C5, wnioskodawca był autorem hipotezy, określił proces badawczy, przygotował kod metody, przeprowadził badania, opracował wyniki, zweryfikował hipotezę oraz przygotował manuskrypt]. Zaproponowane rozwiązanie pozwoliło na lepszą kontrolę wykorzystania zasobów serwera. Implementacja i eksperyment testowy

wykazały mniejsze skoki wykorzystania zasobów serwera w stosunku do metody pierwotnej [C1]. Rozwiązanie dzięki swojej prostocie i małym wymaganiom na moc obliczeniową może być wykorzystywane w małych rozwiązaniach typu IoT, co jest dużą zaletą tej propozycji radzenia sobie z atakami na zasoby sieciowe.

Wreszcie w ramach badań nad metodą, po zastosowaniu sterowania mechanizmem kontroli przepływu pakietów w postaci obliczeń z użyciem skierowanych liczb rozmytych, przeprowadzono *badania wpływu metod wyostrzania na sterowanie* mechanizmem przesyłania pakietów przez zaproponowaną metodę walki z atakami na urządzenia sieciowe, przedstawione w publikacji [C6]. W ramach badań sprawdzono wpływ użycia takich metod wyostrzania wyników jak: Golden ratio, Mandala, FOM, MOM i LOM. Badania wykazały, że w zależności od konkretnej metody wyostrzania, proces sterowania może zachodzić wolniej lub szybciej. Prowadzenie eksperymentów z metodami wyostrzania ma znaczenia dla zastosowania rozwiązań z użyciem logiki rozmytej. Otóż nie opracowano jeszcze jasnych wskazówek jaki wpływ na otrzymane wyniki będzie miał wybór danej metody wyostrzania. Budując więc sterowniki konieczne jest rozpatrzenie na czym zależy projektantowi w przewidzianym zastosowaniu. Konieczne jest jeszcze przeprowadzenie szeregu badań nad metodami wyostrzania w celu opracowania uniwersalnych metod ich doboru do danego zastosowania. Autor [C6] podjął takie prace widząc braki literaturowe w tej materii.

Wreszcie *do zagadnień bezpieczeństwa wykorzystano opracowaną sztuczną sieć neuronową wykorzystującą skierowane liczby rozmyte*, wyniki badań przedstawiono w publikacji [C7]. Opracowaną sieć wykorzystano do wykrywania i predykcji anomalii na bazie dostarczonych logów z działającego systemu. Rozwiązanie porównano z siecią LSTM oraz siecią głęboką. Zaproponowana sieć ze skierowanymi liczbami rozmytymi cechowała się bardzo dużą skutecznością, a przede wszystkich też dużo krótszym czasem uczenia sieci niż sieci LSTM. Warto zauważyć, że zaproponowane rozwiązanie cechuje się też porównywalnymi wynikami, ale przy mniejszej architekturze sieci, co ma wpływ na zasoby niezbędne do jej uczenia i wykorzystania.

W ramach badań nad zagadnieniami bezpieczeństwa rozwiązań, *dokonano również badań z użyciem zaproponowanej sieci rozmytej dotyczących wykrywania anomalii w sieci blockchain,* a wyniki przedstawiono w publikacji [C8, wnioskodawca był autorem hipotezy, współpracował przy określeniu procesu badawczego, przygotowania kodu metody w zakresie znanych metod oraz opracowaniu wyników, następnie zweryfikował hipotezę oraz przygotował manuskrypt]. Opracowaną sieć wykorzystano do wykrywania takich anomalii jak: Sybil Attack, Double-Spending, Smart Contacts, Analysis of Transactions with a Large Number of

Confirmations Equal to Zero, High-Frequency Transaction Anomaly. Rozwiązanie porównano z istniejącymi metodami: Random Forest Classifier, Gradient Boosting Classifier, k-means Algorithm, One-Class SVM (Support Vector Machine), Autoencoder. Uzyskane wyniki wykazały możliwości zastosowania zaproponowanej sieci rozmytej i osiągnięcie porównywalne do metod używanych do tej pory, co pozwala pozytywnie ocenić rozwiązanie do wykrywania anomalii w sieci blockchain. Zaletą rozwiązania jest zastosowanie jednej technologii do wykrywania wybranych zagrożeń. Jedna metoda, to jeden sposób uczenia, wykrywania i jedna biblioteka programistyczna do użycia, co ma istotne znaczenia dla szerokiego wdrożenia technologii blockchain.

Trzecim przedstawionym do oceny osiągnięciem jest oryginalne osiągnięcie projektowe, konstrukcyjne i technologiczne jakim jest współautorstwo rozwiązania HMS C3IS JAŚMIN.

HMS C3IS JAŚMIN jest Systemem Wspomagania Dowodzenia dla Sił Zbrojnych Rzeczypospolitej Polskiej, wdrożonym w nieograniczonej licencji, z unikalnym w skali świata rozwiązaniem bramki/gatewaya łączącego różne standardy protokołu wymiany informacji wojskowych MIP poprzez opracowanie i zastosowanie konwersji danych z wykorzystaniem ontologicznej baz danych. Obecnie jest to system referencyjny dla systemów NATO ze względu na fakt, że posiada unikalne w skali świata rozwiązanie bramki/gatewaya łączącego różne standardy protokołu wymiany informacji wojskowych MIP. Rozwiązanie to zostało opracowane poprzez połączenie nauki (ontologicznych baz danych) z technikami kodowania, dzięki czemu udało się opracować skuteczny i wydajny (pracujący w czasie rzeczywistym) system konwersji danych z wykorzystaniem ontologicznej baz danych. Oryginalność rozwiązania potwierdzają referencje udzielane właścicielowi praw do rozwiązanie – firmie TELDAT Sp. z o.o. sp.k. przez dowództwo SZ RP. Bez rozwiązań Systemu JAŚMIN nie byłoby możliwości połączenia Systemów Wspomagania Dowodzenia wojsk NATO pracujących bardzo często z różnymi wersjami protokołu MIP, który standardowo nie ma zapewnionego mechanizmu konwersji danych pomiędzy wersjami. Autor wniosku uczestniczył czynnie w pracach m.in. grupy roboczej NATO MIP (spotkania odbywają się cztery razy w roku w bazie wojskowej w Greding w Niemczech), dzięki czemu miał wpływ na architekturę nowych rozwiązań i sterowanie prac, aby możliwe był zbudowanie bramy do konwersji danych z zastosowaniem ontologii.
W skład rozwiązania poza konwerterem systemów MIP wchodzi również serwer wymiany danych dla różnych komponentów (np. lądowego i powietrznego) do synchronizacji informacji

o sytuacji na teatrze działań. Rozwiązanie było promowane na V Konferencji Łączności w Sieradzu: Henryk Kruszyński, Tomasz Kosowski, Łukasz Apiecionek, CID Server JAŚMIN, w: Ewolucja wojskowych systemów teleinformatycznych oraz lessons learned w świetle misji pokojowych i stabilizacyjnych : V Konferencja Łączności, 23.04.2014 r. / red. M. Frączek, Adres wydawniczy: Sieradz : Piotr Dela, 2014, Strony: 179-189, ISBN: 978-83-930009-2-0
W pracach nad rozwiązaniem uczestniczyłem jako projektant i programista. Potwierdzeniem udziału w powstaniu rozwiązaniu jest zaświadczenie Prezesa firmy TELDAT Sp. z o.o. sp.k.

5. Informacja o wykazywaniu się istotną aktywnością naukową albo artystyczną realizowaną w więcej niż jednej uczelni, instytucji naukowej lub instytucji kultury, w szczególności zagranicznej.

Odbyłem dwa staże naukowe:
[1] Faculty of Information Systems and Applied Computer Sciences (WIAI) – Otto-Friedrich-Universitat Bamberg, Niemcy; termin: 15.11.2017 – 15.01.2018; czas trwania: 8 tygodni; charakter stażu: prowadzenie badań nad uniwersalną i bezpieczną architekturą dla systemów IoT.
[2] Akademia Marynarki Wojennej w Gdyni, Wydział Mechaniczno-Elektryczny, Instytu Elektrotechniki i Automatyki Okrętowej, Katedra Informatyki; termin stażu: 01.07.2024-30.09.2024; czas trwania: 3 miesiące; charakter stażu: prowadzenie badań nad rozmytymi sieciami neuronowymi, ze szczególnym uwzględnieniem sieci konwolucyjnych.

Odbyłem trzy wizyty studyjne z wykładami w ramach programu Erasmus:
[1] Wizyta studyjna w ramach programu Erasmus na Otto-Friedrich-Universitat Bambergw roku akademickim 2013/2014.
[2] Wizyta studyjna w ramach programu Erasmus na Technical University of Košice w roku akademickim 2015/2016.
[3] Wizyta studyjna w ramach programu Erasmus na Technical University of Košice w roku akademickim 2014/2015.

Wynikiem stażów i wizyt studyjnych jest nawiązanie współpracy badawczej, której efektami są wspólne publikacje naukowe:
- z pracownikami Uniwersytety w Bambergu:

[1] Autorzy: Łukasz Apiecionek, Marcel Grossmann, Udo Krieger, Tytuł: Harmonizing IoT-Architectures with Advanced SecurityFeatures - A Survey and Case Study, Journal of Universal Computer Science - 2019, Vol. 25, no 6, pp. 571-590, p-ISSN: 0948-695x e-ISSN: 0948-6968, 2019, DOI: 10.3217/jucs-025-06-0571

[2] Apiecionek, Ł., Apiecionek, M., Illig, S., & Grossmann, M. (2018). System monitorowania podstawowych warunków środowiska w Bibliotece Państwowej oraz w Bibliotece Uniwersytetu Ottona i Fryderyka w Bambergu. Fides: Biuletyn Bibliotek Kościelnych, 24(2), 93-105.

- z pracownikami Uniwersytetu w Koszycach:

[3] Zarzycki, H., Dobrosielski, W., Apiecionek, Ł., & Vince, T. (2020). Center of Circles Intersection, a New Defuzzification Method for Fuzzy Numbers. Bulletin of the Polish Academy of Sciences. Technical Sciences, 68(2), 185-190. https://doi.org/10.24425/bpasts.2020.131850

[4] Apiecionek, Ł., Sobczak, M., Makowski, W., & Vince, T. (2015). Multi Path Transmission Control Protocols as a security solution. In V. Novitzká, Š. Korečko, & A. Szakál (Eds.), Informatics 2015: IEEE 13th International Scientific Conference on Informatics, November 18-20, Poprad, Slovakia: Proceedings (pp. 27-31). Piscataway: IEEE. https://doi.org/10.1109/Informatics.2015.7377802

Realizowałem również wspólne projekty badawcze z pracownikami: WAT, NASK-PIB, CNBOP-PIB, SGSP, Uniwersytetem Kaliskim, Politechniką Bydgoską. Efektem wspólnych badań są również publikacje naukowe:

- Łubkowski, P., Krygier, J., Sondej, T., Dobrowolski, A. P., Apiecionek, Ł., Znaniecki, W., & Oskwarek, P. (2023). Decision support system proposal for medical evacuations in military operations. Sensors, 23(11), Art. No. 5144. https://doi.org/10.3390/s23115144.

- konferencje:

[1] Prakseologia współdziałania podmiotów odpowiedzialnych za bezpieczeństwo w sytuacjach kryzysowych, 22-23.11.2023, Akademia Kaliska, Komitet Naukowy

[2] III OGÓLNOPOLSKA KONFERENCJA NAUKOWA – PRAKSEOLOGIA W NAUKACH O BEZPIECZEŃSTWIE – Edukacja na rzecz bezpieczeństwa i obronności - 14-15 listopada 2023 r. – przedstawiciel TELDAT – współorganizatora

[3] Prakseologia w sztuce wojennej (18-19.04.2024), Akademia Kaliska – Wojska obrony terytorialnej w kształtowaniu bezpieczeństwa - przedstawiciel TELDAT – współorganizatora

[4] OGÓLNOPOLSKA KONFERENCJA NAUKOWA PRAKSEOLOGIA W NAUKACH O BEZPIECZEŃSTWIE - PRAKSEOLOGIA WSPÓŁDZIAŁANIA PODMIOTÓW ODPOWIEDZIALNYCH ZA BEZPIECZEŃSTWO W SYTUACJACH KRYZYSOWYCH - 22-23.11.2022, członek komitetu naukowego

[5] Prakseologia w sztuce wojennej OGÓLNOPOLSKA KONFERENCJA NAUKOWA nt. Prakseologia w sztuce wojennej. Wojska Lądowe w kształtowaniu bezpieczeństwa, Akademia Kaliska - 08 - 09.03.2023 r. - członek komitetu naukowego

[6] CISIS2020 (13thInternational Conference on Computational Intelligence in Security for Information Systems) – członek komitetu programowego

[7] Wspomaganie procesów zarządzania działaniami w straży pożarnej w 2016.12.15 - członek rady programowo-naukowej konferencji

[8] Innowacyjne technologie w straży pożarnej w 2018.03.15 - członek rady programowo-naukowej konferencji

[9] XXI Konferencji Naukowo Technicznej "Automatyzacja Dowodzenia" - członek komitetu programowego

[10] International Conference on Professor Witold Kosiński's Heritage, Workshop on Ordered Fuzzy Numbers (WOFN) - członek komitetu organizacyjnego

[11] I Konferencji Naukowej Problemy Bezpieczeństwa i Zarządzania Kryzysowego - członek komitetu organizacyjnego

[12] 4rd International Conference on Management Science and Management Innovation - członek komitetu programowego

[13] Konferencja naukowa "Wspomaganie procesu zarządzania działaniami w Straży Pożarnej" - członek komitetu programowego

[14] Konferencja naukowa "Image Processing and Communication 9TH INTERNATIONAL CONFERENCE" - członek komitetu programowego

[15] Konferencja naukowa "The 14th International Conference on P2P, Parallel, Grid, Cloud and Internet Computing", sekcja "Security and Privacy for Distributed Systems" - członek komitetu programowego

[16] 4th International Conference on Computational Intelligent in Security for Information Systems (CISIS 2021) - członek komitetu pogramowego

[17] 13th International Conference on Computational Intelligence in Security for Information Systems sesja: SS01 – Special Session on Fake News Detection and Prevention - członek komitetu programowego

Byłem również promotorem dwóch prac magisterskich realizowanych na zlecenie przemysłu na innej uczelni – Politechnice Bydgoskiej w Bydgoszczy.

6. Informacja o osiągnięciach dydaktycznych, organizacyjnych oraz popularyzujących naukę lub sztukę.

Byłem promotorem pomocniczym w trzech zamkniętych pozytywnie przewodach doktorskich w Uniwersytecie Kaliskim z dziedziny nauk społecznych, dyscypliny nauk o bezpieczeństwie:

- Krzysztof Wosiński: „Znaczenie wywiadu opartego na otwartych źródłach (OSINT) w zapewnieniu bezpieczeństwa systemów teleinformatycznych i bezpieczeństwa osobowego", kierownik naukowy: płk. rez. dr. hab. inż. Piotra Dela,
- Wojciech Znaniecki: „Usprawnienia procesu kierowania projektami badawczo-rozwojowymi, poprawiające skuteczność kierowania projektami badawczo-rozwojowymi realizowanymi w obszarze obronności i bezpieczeństwa państwa", kierownik naukowy: prof. dr hab. inż. Jarosław Wołejszo,
- Robert Palka: „Koncepcja rozwoju Systemu Wspomagania Dowodzenia HMS C3IS JAŚMIN wojsk lądowych w działaniach sojuszniczych i koalicyjnych", kierownik naukowy: prof. dr hab. inż. Jarosław Wołejszo.

Od 2011 roku prowadzę zajęcia dydaktyczne na Uniwersytecie Kazimierza Wielkiego w Bydgoszczy. Przygotowywałem i prowadziłem zajęcia z następujących przedmiotów (wykłady, laboratoria i ćwiczenia):

- Bezpieczeństwo systemów informatycznych,
- Bezpieczeństwo teleinformatyczne,
- Grafika i techniki multimedialne,
- Grafika komputerowa,
- Oprogramowanie systemów zarządzania i BHP,
- Podstawy teleinformatykia,
- Pracownia dyplomowa,
- Seminarium dyplomowe,
- Programowanie aplikacji sieciowych,
- Sieci komputerowe,
- Sieci komputerowe i aplikacje sieciowe,

- Specjalnościowa pracownia dyplomowa,
- Systemy informatyczne w BHP,
- Systemy zarządzania BHP i oprogramowanie,
- Usługi sieciowe,
- Wstęp do sztucznej inteligencji,
- Wybrane zagadnienia sztucznej inteligencji.

Wypromowałem 50 prac inżynierskich oraz 13 magisterskich (w tym dwie na zlecenie przemysłu na innej uczelni oraz jedną w języku angielskim).

Jestem współorganizatorem grupy popularyzującej technologię programowania .Net pod nazwą: Bydgoszcz .Net User Group. Obecnie grupa prowadzi wykłady cykliczne z przedmiotowej technologii.

7. Oprócz kwestii wymienionych w pkt. 1-6, wnioskodawca może podać inne informacje, ważne z jego punktu widzenia, dotyczące jego kariery zawodowej.

Łukasz Apiecionek

……..……..…….
(podpis wnioskodawcy)