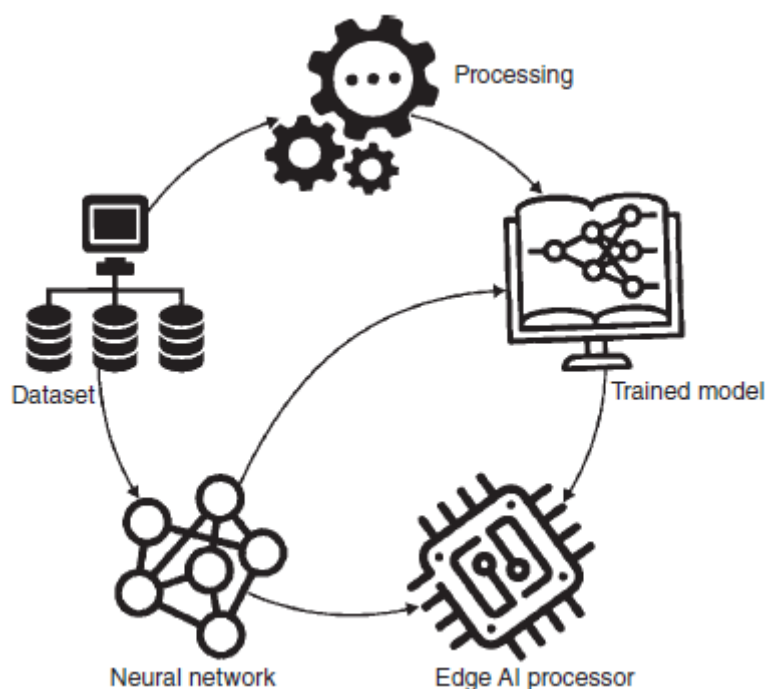


Nowatorskie przenoszenie danych za pomocą głębokiego uczenia się, model wykrywania cyberataków na potrzeby obliczeń brzegowych w sieciach 6G

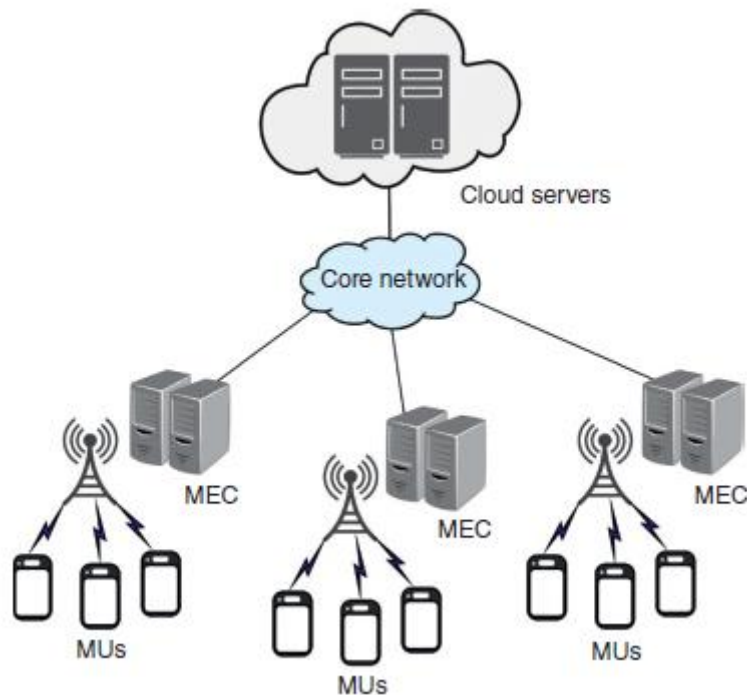
Wstęp

Ponieważ 5G przyniosło pionową transformację w celu zmiany społeczeństwa i stopniowo otworzyło kurtynę Internetu wszystkiego, 6G otworzyło innowacyjną erę „Internetu inteligencji” z połączonymi rzeczami, inteligencją i ludźmi, rozwiązując ludzkie problemy w różnych aspektach. Aby autoryzować sieć 6G wykorzystując możliwości sztucznej inteligencji (AI), Internet rzeczy (IoT) i urządzenia mobilne generowałyby w nieskończoność ogromną ilość informacji multimodalnych (na przykład filmów użytkowników, zapisów zachowań i plików audio) o środowiskach fizycznych. Gadżety znajdujące się na brzegu sieci. Kierując się tym trendem, istnieje pilna potrzeba przesunięcia granic sztucznej inteligencji na brzeg sieci, aby całkowicie uwolnić sieć 6G. Aby zaspokoić potrzeby użytkowników, przetwarzanie brzegowe – wyłaniający się model, który przenosi usługi i zadania obliczeniowe z rdzenia sieci na brzeg sieci – było powszechnie znane jako niezbędny element przyszłej sieci 6G. Mobile Edge Computing (MEC) to alternatywny system transmisji, który świadczy usługi użytkownikom. To narzędzie umożliwia model przetwarzania mobilnego z autonomicznym serwerem i funkcjami uzyskiwanymi z warstw IoT lub chmur. Zasób pochodny udostępniany jest użytkownikom w celu świadczenia usług kompleksowych. Obiekt obejmuje mobilność, transformację wsparcia, elastyczność, niezawodność i możliwości adaptacji, które obsługują różną gęstość urządzeń użytkownika. Rysunek ilustruje moduły AI z obliczeń brzegowych.



Żądanie obliczeniowe jest usprawniane z warstwy brzegowej do warstwy górnej, w tym IoT lub chmury. W ramach IoT brane są pod uwagę różne aplikacje i udogodnienia związane ze frameworkami mgły i krawędzi, aby generować skalowalne sieci transmisyjne z analizą i obliczeniami danych. Zadanie przesyłania danych od użytkowników końcowych do MEC w celu przeprowadzenia dalszych obliczeń jest szeroko badane na podstawie istniejących badań podczas badania ograniczeń transmisji i obliczeń [7]. Liczne podejścia i metody odciążania obliczeń w MEC zostały zaawansowane technologicznie na podstawie badań w celu przypisania wykorzystania energii, zmniejszenia opóźnień obliczeniowych i efektywnego wykorzystania zasobów radiowych. Osiąganie lepszych wyników w dynamicznych i

złożonych bezprzewodowych schematach MEC dla wielu użytkowników to skomplikowane procedury. Ponadto zagrożenia bezpieczeństwa związane z przesyłaniem informacji są eliminowane każdą metodą odciążania. Poza tym nieodpowiednie kontrole bezpieczeństwa danych mogą szybko przewyższyć korzyści płynące z techniki MEC. Aby przeciwdziałać zagrożeniu cybernetycznemu w MEC, kluczowe może być rozróżnienie wcześniejszych cyberataków, a tym samym podjęcie szybszych środków zaradczych w celu uniknięcia ryzyka. Rysunek przedstawia infrastrukturę MEC.



Badacze zaprezentowali metodę matematyczną całkowitego opóźnienia usług schematu MEC opartego na cyberbliźniaku, która obejmuje migrację serwerów wirtualnych, licznych serwerów fizycznych, mobilność użytkowników na różnych poziomach sieci, żądanie treści/buforowanie, przetwarzanie, backhaul i fronthaul komunikacji oraz przedstawił algorytm kierujący procesem cyberbliźniaków i płaszczyzną sterowania z sytuacji MEC. Liao i inni przedstawili rozproszoną, dwustopniową metodę odciążania (DTSO) zapewniającą rozwiązanie kompromisowe. Początkowo, poprzez przedstawienie koncepcji kolejkowania i uwzględnienie zakłóceń kanałów, generowane jest zagadnienie optymalizacji kombinacyjnej w celu obliczenia możliwości odciążenia wszystkich stacji. Następnie nowatorskie zagadnienie przekształca się w nieliniowo zoptymalizowany problem, który jest rozwiązywany za pomocą modelu sekwencyjnego programowania kwadratowego (SQP). Uwzględniono schemat MEC wykorzystujący ograniczony bufor obliczeniowy na serwerze brzegowym. Na schemacie procedura obliczeniowa i komunikacyjna tworzą pętlę sprzężenia zwrotnego i nie można ich oddzielić. Następnie zaprezentowano technikę dwufazowego kolejkowania tandemowego w czasie dyskretnym. Dai i inni przedstawiają rekonfigurowalną inteligentną powierzchnię (RIS) do przetwarzania brzegowego w celu obsługi aplikacji o niższych opóźnieniach, w których przetwarzanie brzegowe mogłoby poprawić większą presję obliczeniową urządzeń mobilnych z uniwersalnie rozproszonymi zasobami obliczeniowymi, a RIS poprawia jakość bezprzewodowego połączenia transmisyjnego poprzez percepcyjną zmianę środowiska propagacji radiowej. W tym badaniu opracowano nowatorski model odciążania danych z wykorzystaniem głębokiego uczenia się i wykrywania cyberataków (DADL-CAD) na potrzeby przetwarzania brzegowego w sieciach 6G. Proponowana technika DADL-CAD projektuje przede wszystkim model rekurencyjnej sieci neuronowej (RNN) do prognozowania przepływu ruchu w sieciach 6G obsługujących przetwarzanie brzegowe.

Wykorzystano także model adaptacyjnej entropii krzyżowej próbkowania (ASCE) w celu maksymalizacji wydajności sieci poprzez właściwe podejmowanie decyzji związanych z procesem odciążania. Ponadto przeprowadzono konkurencyjną optymalizację roju (CSO) za pomocą techniki stosowego autoenkodera (SAE) w celu wykrycia cyberataków w sieci. Walidacja wydajności techniki DADL-CAD jest badana w różnych aspektach.

Proponowany model

W tym badaniu opracowano nowe podejście DADL-CAD do przesyłania danych i wykrywania cyberataków w sieciach 6G. Proponowana technika DADLCAD obejmuje szereg procesów, a mianowicie prognozowanie ruchu w oparciu o RNN, przesyłanie danych w oparciu o ASCE, wykrywanie cyberataków w oparciu o SAE oraz dostrajanie parametrów w oparciu o CSO. Model ASCE wykorzystywany jest do maksymalizacji efektywności sieci poprzez właściwe podejmowanie decyzji związanych z procesem odciążenia

Prognozowanie przepływu ruchu w oparciu o RNN

W początkowej fazie do prognozowania potoku ruchu można wykorzystać model RNN. Notacja dla pojedynczego przypadku badawczego jest ponadto wyraźnie równoważna serii długości.

$$h_1 = \tanh (W_{xh}x_1 + b_h), \quad (2.1)$$

$$p(\gamma_1 | x_1) = \sigma(W_{hy}h_1 + b_y). \quad (2.2)$$

Następnie założmy, że zmiana transformacji liniowej wykorzystanej na podstawie obliczeń stanu ukrytego była zależna nie tylko od naszych danych wejściowych x , poza danymi z przeszłości, które są obsługiwane przez stan ukryty. Ustawiliśmy nasz poprzedni stan ukryty na 0, co jest rozumiane jako nie obsługujące żadnych danych w przeszłości:

$$h_0 = 0, \quad (2.3)$$

$$h_1 = \tanh (W_{hh}h_0 + W_{xh}x_1 + b_h), \quad (2.4)$$

$$p(\gamma_1 | x_1) = \sigma(W_{hy}h_1 + b_y). \quad (2.5)$$

Warto zauważyć, że z perspektywy modelowania sieć ta była w szczególności równa sieci nowatorskiej; różni się jedynie zapisem i licznikiem funkcji (przedstawianym przez niepotrzebną mnożnik macierzowo-wektorowy zawierający $h_0 = 0$). Jednak sieć jest oczywiście rozszerzona na pacjentów z pewną liczbą badań:

$$h_0 = 0, \quad (2.6)$$

$$h_1 = \tanh (W_{hh}h_0 + W_{xh}x_1 + b_h), \quad (2.7)$$

$$h_2 = \tanh (W_{hh}h_1 + W_{xh}x_2 + b_h), \quad (2.8)$$

$$h_T = \tanh (W_{hh}h_{T-1} + W_{xh}x_T + b_h), \quad (2.9)$$

$$p(\gamma_T | x_1, \dots, x_T) = \sigma(W_{hy}h_T + b_y). \quad (2.10)$$

RNN był w stanie przetworzyć szeregi o niektórych długościach, tj. jeden o długości $T = 1$ i drugi o długości $T = 7$, ponieważ operacja przejścia i ich parametry są wspólne w czasie. Jeżeli zostało to

określone, szkolenie sieci zostało wdrożone w oparciu o podejście bardzo podobne do procedury wyjaśnionej dla sieci ze sprzężeniem zwrotnym. Wszyscy pacjenci odpowiadają kolejności badań x_1, \dots, x_T wraz z etykietą y_T . Może to być procedura obliczania wykresu z rozwinięciem RNN w tych przedziałach czasowych i dodaniem funkcji obliczających prawdopodobieństwo nowotworu złośliwego i straty. Następnie gradient można uzyskać stosując propagację wsteczną (BP) i optymalizując wykorzystanie stochastycznego opadania gradientowego (SGD). Początkowe odmiany RNN ustalono na podstawie literatury dotyczącej łatwych RNN lub Elmana RNN. W analizach klasycznie rozumie się dalszą zwartą reprezentację:

$$h_t = \tanh(W_{hh}h_{t-1} + W_{xh}x_t + b_h) \quad (2.11)$$

podczas gdy pierwotny stan ukryty został pominięty i, jeśli nie został zidentyfikowany, często uważa się, że $h_0 = 0$. Chociaż uproszczony wykres obliczeniowy jest równoważny równaniu. (2.11).

Przesyłanie danych w oparciu o ASCE

Model ASCE wykorzystywany jest do maksymalizacji efektywności sieci poprzez właściwe podejmowanie decyzji związanych z procesem odciążenia. Jest to najbardziej binarny problem programowania liczb całkowitych, który można skutecznie rozwiązać za pomocą techniki rozgałęziania i wiązania (BnB), co wiąże się z ogromnymi trudnościami w przetwarzaniu, szczególnie gdy X jest lepszy. Ponadto ogromna liczba operacji usprawnia podejście do obliczeniowych punktów dostępu (CAP). Następnie technologie BnB spełniają zapotrzebowanie na funkcje praktyczne. Następnie w pracy próbuje się rozwiązać problem, wdrażając typowe zoptymalizowane techniki. Godnym uwagi rezultatem jest zastosowanie relaksacji wypukłej, takiej jak relaksacja $x_{nm} \in \{0, 1\}$ jako $x_{nm} \in [0, 1]$ poprzez relaksację programowania liniowego (LPr):

$$T(X) = \max_{m \in \mathcal{M}} T_m(X) \text{ as } T(X) \geq \max_{m \in \mathcal{M}} T_m(X) \quad (2.12)$$

Jednak relaksacja prowadzi do pogorszenia wydajności, jeśli jest związana z techniką BnB. Następnie rozwiązano dyskretny zoptymalizowany parametr, wykonując podejście probabilistyczne na podstawie poznania prawdopodobieństw polityki x_{nm} . Aby rozwiązać ten problem, zastosowano technologię entropii krzyżowej (CE) wraz z próbkowaniem adaptacyjnym jako ASCE.

Wykrywanie cyberataków w oparciu o SAE

Autoenkoder to rodzaj modelu uczenia się bez nadzoru, który obejmuje warstwy wyjściowe, wejściowe i ukryte. Procedura uczenia autoenkoderów (AE) obejmuje kodowanie i dekodowanie. Do mapowania wejściowego zbioru danych na obraz ukryty wykorzystano kodowanie, a dekodery służą do odtworzenia wejściowego zbioru danych na obraz ukryty. Załóżmy, że nieoznakowane dane wejściowe $\{x_n\}_{n=1}^N$, gdzie $x_n \in \mathbb{R}^{m \times 1}$, h_n oznacza ukryty wektor kodowania obliczony z x_n , a \hat{x}_n oznacza wektor dekodujący warstwy wynikowej:

$$h_n = f(W_1 x_n + b_1) \quad (2.13)$$

W którym f reprezentuje funkcję kodera, W_1 wskazuje macierz wag, a b_1 pokazuje wektor obciążenia:

$$\hat{x}_n = g(W_2 h_n + b_2) \quad (2.14)$$

Tutaj g wskazuje funkcję dekodera, W_2 pokazuje macierz wag, a b_2 wskazuje wektor odchylenia. Zmienny zestaw AE został ulepszony w celu zminimalizowania błędu rekonstrukcji:

$$\theta(\Theta) = \arg \min_{\theta, \theta'} \frac{1}{n} \sum_{l=1}^n L(x^l, \hat{x}^l) \quad (2.15)$$

Podczas gdy L oznacza funkcję straty, $L(x, \hat{x}) = \|x - \hat{x}\|^2$, architektura SAE składa się z n autoenkodera w n warstw ukrytych za pomocą modelu uczenia się bez nadzoru i dostrajania przez model nadzorowany. Dlatego metodę opartą na SAE klasyfikuje się w następujący sposób:

- 1) Trenowanie początkowego AE poprzez dane wejściowe i osiągnięcie wyuczonego wektora cech.
- 2) Wektor cech poprzedniej warstwy został wykorzystany jako dane wejściowe dla kolejnej warstwy i proces ten jest powtarzalny aż do zakończenia uczenia.
- 3) Następnie trenowana jest każda warstwa ukryta i wykorzystywana jest metoda BP w celu minimalizacji funkcji kosztu i ulepszania wagi za pomocą wyszkolonego zestawu etykiet w celu dokładnego dostrojenia.

Optymalizacja parametrów w oparciu o CSO

Aby dostroić wartości wagi i błędu systematycznego techniki SAE, zastosowano algorytm CSO. CSO to obecne podejście oparte na inteligencji roju, stymulowane optymalizacją roju cząstek (PSO) podejście, ale koncepcja ta znacznie różni się od standardowego PSO [18]. Rozważ n ilości cząstek, $S(t)$ reprezentuje inicjalizację roju. Wszystkie cząstki oznaczają możliwe rozwiązanie. Rój $S(t)$ ma n cząstek, we wszystkich iteracjach $n/2$ par jest przydzielanych arbitralnie, a następnie odbywa się rywalizacja pomiędzy dwiema cząstkami ze wszystkich par cząstek. Ze względu na konkurencję cząstki mają optymalną wartość sprawności, określaną jako „zwycięzca” i zostaną bezpośrednio przekazane do kolejnej iteracji roju, $S(t + 1)$, natomiast cząstka, która przegrywa konkurencję, nazywana jest „przegraną”, „poprawiłyby swoją pozycję i prędkość, ucząc się od zwycięzcy. We wszystkich iteracjach cząstka raz bierze udział w konkursie. Rozważ pozycję i prędkość zwycięzcy i przegranego w m -tej rundzie zawodów w iteracji t z odpowiednio $V_{w,m}(t)$, $V_{l,m}(t)$ i $X_{w,m}(t)$, $X_{l,m}(t)$, natomiast $m = 1, 2, 3, \dots, n/2$. Zgodnie z podstawową koncepcją CSO, po rundzie zawodów pozycja i prędkość przegranego zostaną ulepszone w następujący sposób:

$$V_{l,m}(t + 1) = r_1(m, t) \times V_{l,m}(t) + r_2(m, t) \times (X_{w,m}(t) - X_{l,m}(t)) + \varphi \times r_3(m, t) \times (\bar{X} - X_{l,m}(t)) \quad (2.16)$$

$$X_{l,m}(t + 1) = X_{l,m}(t) + V_{l,m}(t + 1) \quad (2.17)$$

W którym $r_1(m, t)$, $r_2(m, t)$ i $r_3(m, t) \in [0, 1]$ oraz $\bar{X}(t)$ oznaczają średnią lokalizację i są określane na dwa sposoby, średnia globalna i średnia lokalna reprezentowana przez $\bar{X}_m^g(t)$ - i $X_l^g(t)$. $X_m^g(t)$ oznaczają globalną średnią lokalizację każdej cząstki, gdzie $\bar{X}_l^g(t)$ - oznacza lokalną średnią z określonego sąsiedztwa cząstki k . φ oznacza zmienne kontrolujące wpływ $\bar{X}_m(t)$. Aby lepiej zrozumieć GUS, w badaniu przedstawiono powiązane porównanie z wykorzystaniem PSO:

1) Część początkowa $r_1(m, t) \times V_{l,m}(t)$ jest analogiczna do składnika bezwładności w PSO, który równoważy ruch roju, wariacją jest waga bezwładności w , która jest wymieniana jako dowolny wektor $r_1(m, t)$ w GUS.

2) Następną część $r_2(m, t) \times (X_{w, m}(t) - X_{l,m}(t))$ jest analogiczna do elementu poznawczego w PSO, ale teoretycznie jest całkiem odrębna w PSO, cząstka traci konkurencję od zwycięzcy, a nie od rekordu życiowego znalezionej do tej pory. Ta metoda jest bardziej wykonalna biologicznie, gdy inspirowane zachowanie roju, ponieważ trudno jest zapamiętać własne najlepsze doświadczenia.

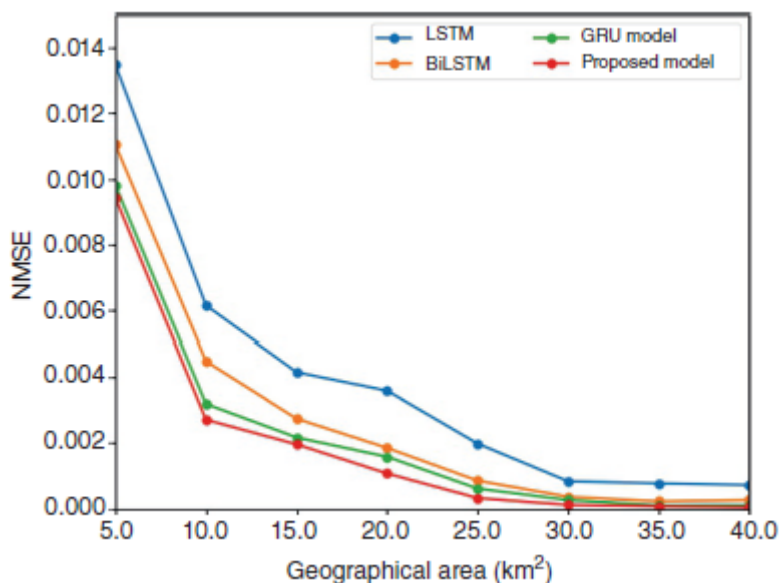
3) Ostatnia część $r_3(m, t) \times (\bar{X}(t) - X_{l,m}(t))$ jest analogiczna do elementu społecznego w standardowym PSO, ale cząstki tracą konkurencję wynikającą ze średniej lokalizacji, a raczej gbest w PSO; nie jest potrzebna pamięć, co mogłoby być biologicznie bardzo wykonalne.

Do potwierdzenia jakości cząstek wykorzystano sprzężenie zwrotne (FF). Celem CSO jest odkrycie położenia cząstek, które skutkuje optymalną oceną funkcji przystosowania. W metodzie inicjalizacji CSO wszystkim cząstkom przypisuje się dowolną prędkość i położenie. PSO ustanowiło standardy w różnych zastosowaniach, w tym w problemach optymalnego wyboru funkcji w kosztach oprogramowania, algorytmach uczenia maszynowego, routingu, klastrowaniu i tak dalej. CSO uzyskało w testowanej funkcji standardowej optymalny wynik niż PSO, stąd motywuje nas to do zastosowania w przedstawionej metodzie problemu wyboru i routingu głowicy klastra (CH).

Walidacja wydajności

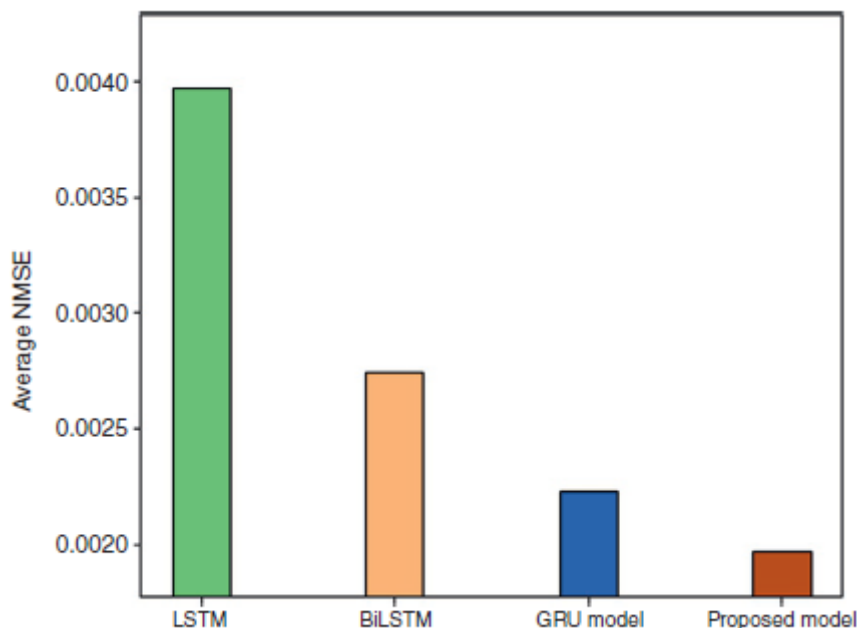
W tej części dokonano analizy wyników techniki DADL-CAD pod kątem różnych aspektów. Po pierwsze, analiza znormalizowanego błędu średniokwadratowego (NMSE) algorytmu DADL-CAD z najnowszymi algorytmami DL ma miejsce w tabeli i rysunku.

Area (km ²)	LSTM	BiLSTM	GRU model	Proposed model
5	0.013 48	0.011 05	0.009 81	0.009 43
10	0.006 15	0.004 49	0.003 20	0.002 72
15	0.004 16	0.002 75	0.002 18	0.001 97
20	0.003 60	0.001 86	0.001 59	0.001 08
25	0.001 99	0.000 86	0.000 62	0.000 33
30	0.000 84	0.000 37	0.000 27	0.000 12
35	0.000 78	0.000 23	0.000 10	0.000 08
40	0.000 73	0.000 27	0.000 10	0.000 05
Average	0.003 97	0.002 74	0.002 23	0.001 97

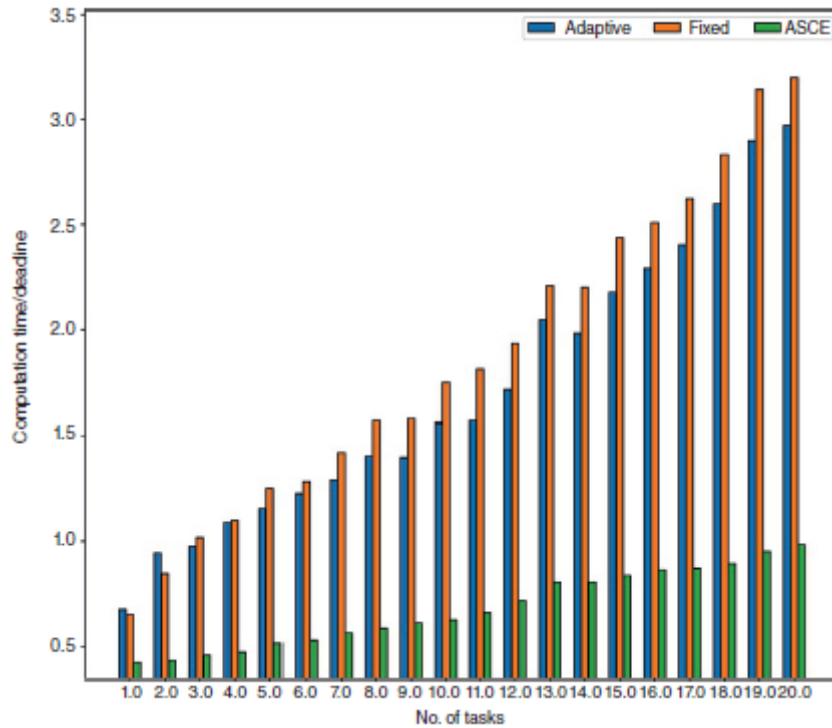


Wyniki wykazały, że algorytm DADL-CAD doprowadził do obniżenia wartości NMSE w różnych obszarach. Na przykład dla obszaru 5 km² algorytm DADL-CAD zaoferował niższy NSME wynoszący 0,009 43, podczas gdy algorytmy LSTM, BiLSTM i GRU osiągnęły maksymalne NMSE wynoszące odpowiednio 0,001 348, 0,011 05 i 0,009 81. Ponadto przy powierzchni 10 km² algorytm DADL-CAD zaoferował niższy NMSE wynoszący 0,006 15, podczas gdy algorytmy LSTM, BiLSTM i GRU osiągnęły maksymalne NMSE wynoszące odpowiednio 0,006 15, 0,004 49 i 0,003 20. Co więcej, przy powierzchni 15 km² algorytm DADL-CAD zapewnił minimalne NMSE wynoszące 0,004 16, podczas gdy algorytmy LSTM, BiLSTM i GRU uzyskały wyższe NMSE wynoszące odpowiednio 0,002 75, 0,002 18 i 0,001 97. Co więcej, na obszarze 20 km² algorytm DADL-CAD oferuje niższy znormalizowany średni błąd umiarkowany (NMME) wynoszący 0,001 99, podczas gdy algorytmy LSTM, BiLSTM i GRU osiągnęły maksymalne NMSE wynoszące 0,000 82, 0,000 62 i 0,000 33, odpowiednio.

Rysunek przedstawia średnią analizę NSME modelu DADLCAD przy użyciu najnowszych metod.



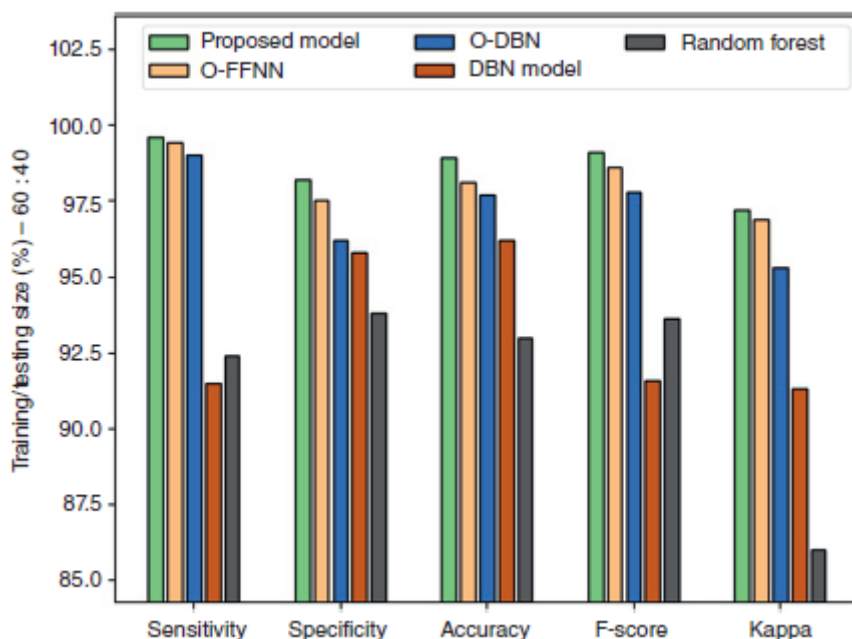
Wyniki wykazały, że model DADL-CAD osiągnął efektywne wyniki przy minimalnym średnim NMSE wynoszącym 0,003 97, ale techniki LSTM, BiLSTM i GRU osiągnęły wyższe średnie NMSE wynoszące 0,002 74, 0,002 23 i odpowiednio 0,001 97. Rysunek 2.5 przedstawia analizę całkowitego czasu realizacji zadania metodą DADL-CAD.



Wyniki wykazały, że technika ASCE uzyskała krótszy czas realizacji zadań w porównaniu z modelami adaptacyjnymi i stałymi. Ponadto model ASCE oferował niższy średni czas wykonania zadania wynoszący 0,680, podczas gdy modele adaptacyjne i stałe uzyskały wyższy średni czas wykonania zadania wynoszący odpowiednio 1,718 i 1,867. Skuteczność wykrywania cyberataków metodą DADL-CAD bada się za pomocą najnowszych metod na różnych rozmiarach danych szkoleniowych i testowych. Wyniki wskazują, że model DADL-CAD pozwolił na uzyskanie skutecznych wyników klasyfikacji, jak pokazano w tabeli .

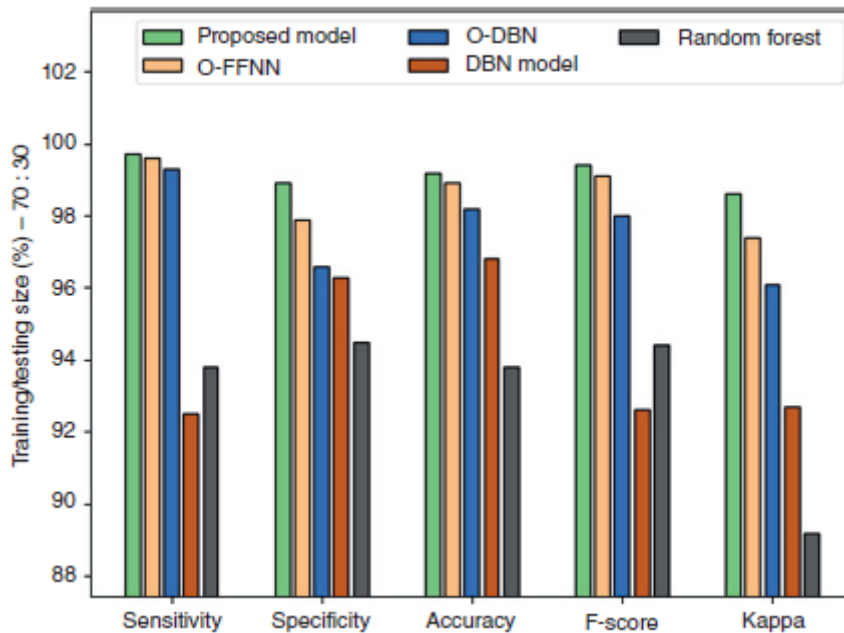
Methods	$Sens_y$	$Spec_y$	Acc_y	F_{score}	Kappa
Training size = 60% testing size = 40%					
Proposed model	99.60	98.20	98.90	99.10	97.20
O-FFNN	99.40	97.50	98.10	98.60	96.90
O-DBN	99.00	96.20	97.70	97.80	95.30
DBN model	91.50	95.80	96.20	91.60	91.30
Random forest	92.40	93.80	93.00	93.60	86.00
Training size = 70% testing size = 30%					
Proposed model	99.70	98.90	99.20	99.40	98.60
O-FFNN	99.60	97.90	98.90	99.10	97.40
O-DBN	99.30	96.60	98.20	98.00	96.10
DBN model	92.50	96.30	96.80	92.60	92.70
Random forest	93.80	94.50	93.80	94.40	89.20

Rysunek przedstawia wyniki wykrywania cyberataków za pomocą modelu DADL-CAD przy użyciu innych metod na danych szkoleniowych/testowych 60:40.



Wyniki wskazują, że model DADL-CAD dał skuteczne wyniki pod względem różnych miar. Jeśli chodzi o czułość, model DADL-CAD uzyskał lepszą czułość o 99,60%, podczas gdy modele O-FFNN, O-DBN, DBN i RF uzyskały niższą czułość wynoszącą odpowiednio 99,40, 99, 91,50 i 92,40%. Co więcej, jeśli chodzi o akcelerację, model DADL-CAD uzyskał lepszą akcelerację wynoszącą 98,90%, podczas gdy modele O-FFNN, O-DBN, DBN i RF uzyskały niższą akcelerację wynoszącą odpowiednio 98,10, 97,70, 96,20 i 93%. Co więcej, jeśli chodzi o F_{score} , algorytm DADL-CAD uzyskał lepszy F_{score} o 99,10%, podczas gdy algorytmy O-FFNN, O-DBN, DBN i RF uzyskały niższy F_{score} wynoszący odpowiednio 98,60, 97,80, 91,60 i 93,60%.

Rysunek ilustruje wyniki wykrywania cyberataków metodą DADLCAD przy użyciu innych metod na danych szkoleniowych/testowych 70:30.



Wyniki pokazały, że model DADL-CAD dał skuteczne wyniki pod względem różnych miar. Jeśli chodzi o $sens_y$, algorytm DADL-CAD uzyskał lepszą $sens_y$ czułość o 99,70%, podczas gdy modele O-FFNN, O-DBN, DBN i RF uzyskały niższą $sens_y$ wynoszącą odpowiednio 99,60, 99,30, 92,50 i 93,80%. Ponadto, jeśli chodzi o acc_y , technika DADL-CAD uzyskała lepszą acc_y wynoszącą 99,20%, podczas gdy systemy O-FFNN, O-DBN, DBN i RF uzyskały niższą acc_y wynoszącą odpowiednio 98,90, 98,20, 96,80 i 93,80%. Wreszcie, jeśli chodzi o F_{score} , technika DADL-CAD uzyskała wyższy F_{score} wynoszący 99,40%, podczas gdy techniki O-FFNN, O-DBN, DBN i RF osiągnęły niższy F_{score} odpowiednio 99,10, 98, 92,60 i 94,40%.

Wniosek

W tym badaniu opracowano nową technikę DADL-CAD do przesyłania danych i wykrywania cyberataków w sieciach 6G. Proponowana technika DADL-CAD obejmuje szereg procesów, a mianowicie prognozowanie ruchu w oparciu o RNN, przesyłanie danych w oparciu o ASCE, wykrywanie cyberataków w oparciu o SAE oraz dostrajanie parametrów w oparciu o CSO. Model ASCE wykorzystywany jest do maksymalizacji wydajności sieci poprzez właściwe podejmowanie decyzji związanych z procesem ociążenia. Walidacja wydajności techniki DADL-CAD jest badana pod różnymi względami, a badanie porównawcze wykazało wyższość techniki DADL-CAD nad najnowszymi podejściami. Dlatego technikę DADL-CAD można uznać za skuteczne narzędzie poprawy wydajności sieci 6G obsługujących MEC. W przyszłości można opracować modele agregacji danych, aby ograniczyć proces komunikacji w sieciach 6G