Prof. dr hab. inż. Krzysztof Wawryn
Wydział Elektroniki i Informatyki
Politechnika Koszalińska
ul. Śniadeckich 2, 75-453 Koszalin

Koszalin, 22 maja 2024 roku

Recenzja rozprawy doktorskiej
pt. "*Sprzętowa implementacja sieci LSTM*",
której autorem jest mgr inż. Grzegorz Rafał Dec

Przedstawiona do recenzji rozprawa doktorska dotyczy sprzętowej implementacji wielowarstwowej rekurencyjnej sieci neuronowej LSTM wykorzystującej głębokie uczenie maszynowe do przetwarzania informacji sekwencyjnych. Charakteryzuje się pamięcią długoterminową i jest wykorzystywana w poszukiwaniu rozwiązań wymagających analizy złożonych danych sekwencyjnych pochodzących z przeszłości. Z tego względu prace z zakresu sztucznych sieci neuronowych umożliwiających głębokie uczenie maszynowe posiadają aktualnie duże zapotrzebowanie w automatyzacji rozwiązywania złożonych zadań niekoniecznie technicznych. Przetwarzanie tych zadań przez sieci LTSM na komputerze ogólnego przeznaczenia z procesorem CPU zajmuje dużo czasu, ponieważ operacje są wykonywane szeregowo. Także komputery wielordzeniowe z procesorem GPU dedykowane do przetwarzania grafiki mogą niedostatecznie szybko przetwarzać zadania wykonywane przez sieci LSTM. Aktualnie prace badawcze nad sieciami LTSM wykonującymi zadania w krótszym czasie, zmierzają do ich realizacji sprzętowej przetwarzającej sygnały równolegle. Takie możliwości dostarczają reprogramowalne układy FPGA. Prace nad projektowaniem i implementacją rekurencyjnych sieci LTSM zbudowanych z setek, a nawet tysięcy komórek LSTM w układach FPGA są prowadzone od lat 90-tych XX wieku wraz z rozwojem technologii nanometrowych i nadal należą do pionierskich. Dlatego podjęte przez Autora zadanie sprzętowej implementacji sieci LSTM jest uzasadnione.

Rozprawa napisana jest na 201 stronach tekstu, zilustrowana 103 rysunkami i 32 tabelami, opatrzona 127 pozycjami wykazu literatury światowej oraz 4 pozycjami prac bezpośrednio związanych z rozprawą (w tym w trzy samodzielne i jedna współautorska z promotorem). Tekst pracy jest podzielony na pięć rozdziałów. Rozdziały pierwszy i drugi zawierają wprowadzenie do omawianych zagadnień, a piąty podsumowanie. Zasadnicza część pracy dotycząca autorskich rozwiązań zawarta jest w rozdziałach trzecim i czwartym. Dodatkowo praca zawiera streszczenia w języku polskim i angielskim. Treść pracy odpowiada tytułowi. Autor w kolejnych rozdziałach opisuje problemy związane z projektowaniem i implementacją rekurencyjnych sztucznych sieci neuronowych LSTM w układach reprogramowalnych FPGA, wprowadza własne rozwiązania, weryfikuje je wskazując na ich praktyczną przydatność poprzez porównanie z sieciami implementowanymi za pomocą procesorów CPU i GPU. Osiągnięcia Autora omówię w trakcie charakteryzowania poszczególnych rozdziałów pracy.

W rozdziale pierwszym Autor bardzo zwięźle uzasadnił potrzebę zajęcia się implementacją rekurencyjnych sieci neuronowych LSTM w strukturach układów reprogramowalnych FPGA

WPŁYNĘŁO
03. CZE. 2024

oraz określił cele i tezy pracy. Cele pracy przedstawione na stronie 12 zakładają opracowanie nowej metody implementacji sieci LSTM w układach FPGA wydajniejszej pod względem czasu obliczeń od implementowanej w układzie z procesorem GPU, oraz weryfikację proponowanej metody w zadaniu klasyfikacji w wybranym procesie przemysłowym. Tezy pracy zostały wymienione na stronie 13. Sprowadzają się do stwierdzenia, że sprzętowa realizacja sieci LSTM w układzie FPGA charakteryzuje się większą szybkością działania oraz umożliwia istotne skrócenie czasu uczenia sieci w porównaniu do realizacji programowej za pomocą procesorów CPU i GPU. Prace badawcze związane z osiągnięciem celów i udowodnieniem postawionych tez Autor przedstawił w kolejnych rozdziałach rozprawy.

W rozdziale drugim stanowiącym wprowadzenie Autor w sposób jasny i klarowny wprowadził i objaśnił badane zagadnienia uczenia maszynowego w sieciach LSTM. W swoich pracach odwołuje się do aktualnych badań dotyczących tej tematyki, ma rozeznanie w ostatnich osiągnięciach naukowych z tej dziedziny, ma orientację w zagadnieniach sztucznej inteligencji. We wprowadzeniu opisał również narzędzia i technologie stosowane w projektowaniu implementacji sieci LSTM. Część opisów jest zwarta, a część zbyt obszerna. Do pierwszej grupy zaliczam podrozdziały 2.1. oraz 2.4.÷2.7. dotyczące użytego oprogramowania i sprzętu, oraz platform ZYNQ, Xilinks ZYNQ i Xilinks Versal, a także narzędzi projektowych układów FPGA. Do drugiej zaliczam podrozdziały 2.2. i 2.3. dotyczące układów FPGA oraz języków opisu sprzętu. Schematy blokowe i ideowe różnych architektur układów reprogramowalnych oraz opisy elementów języka Verilog są informacjami katalogowymi niepotrzebnie powiększającymi o 16 stron treść wprowadzenia i tak obszernej pracy.

Oryginalny dorobek Autora został zawarty w rozdziałach trzecim i czwartym rozprawy. W podrozdziałach 3.1-3.4 Autor opisał schemat komórki LSTM i podał definicje analityczne elementów składowych komórki nazywanych bramkami: zapominającą, wejściową, wyjściową, aktywacji wejścia i aktywacji wyjścia. Przyjął dokładność funkcji aktywacji na poziomie 10^{-7} i porównał znane z literatury rozwiązania implementacyjne funkcji aktywacji w strukturach FPGA. Szczegółowo przebadał trzy rodzaje implementacji: z wykorzystaniem algorytmu CORDIC, wielomianów Czebyszewa oraz zwykłych wielomianów. Dla każdej z implementacji Autor podał liczbę użytych tablic LUT, przerzutników FF oraz jednostek DSP, liczbę cykli na operację, liczbę taktów zegara i maksymalną częstotliwość zegara, a dla implementacji wielomianowych dodatkowo stopień wielomianu oraz czas obliczeń. Analiza otrzymanych wyników wraz z wnioskami została przedstawiona na stronach 80÷83. Do dalszych rozważań zostały przyjęte cztery implementacje po jednej za pomocą algorytmu CORDIC i wielomianów Czebyszewa oraz dwie wersje zwykłych wielomianów drugiego i czwartego stopnia. Zostały one przedstawione w tabeli 3.13 jako warianty 1÷4. Ponadto Autor zestawił je z implementacjami znanymi z literatury w tabeli 3.14. i porównał. Wyciąganie wniosków ilościowych na podstawie tych zestawień jest mało wiarygodne, ponieważ w większości literaturowych wyników brakuje informacji o danych implementacyjnych, a więc nie można stwierdzić czy eksperymenty były przeprowadzane w takich samych bądź zbliżonych warunkach. W implementacji komórki LSTM wykorzystano kolejno cztery wcześniej wybrane implementacje funkcji aktywacji. Na ich podstawie wyciągnięto wnioski, że komórka LSTM z iteracyjną implementacją algorytmu CORDIC jest najwolniejsza, a najlepsze rezultaty z względu na czas przetwarzania można osiągnąć za pomocą komórki LSTM z funkcją aktywacji z aproksymacją za pomocą zwykłych wielomianów. W pracy brakuje opracowania matematycznego obliczania czasu przetwarzania dla wszystkich struktur. Należy także zaznaczyć, że iteracyjna implementacja algorytmu

CORDIC nie jest optymalna, ponieważ najkrótszy czas przetwarzania osiąga się w architekturze potokowej, a nie iteracyjnej.

W podrozdziałach 3.5 i 3.6 Autor przedstawia przykład rozwiązania zadania klasyfikacji wyrobów w procesie kucia na zimno za pomocą sieci LSTM w układach FPGA i w układach z procesorami ARM z elementami logiki programowalnej projektowanych za pomocą platformy ZYNQ. Przykład zadania klasyfikacji kucia na zimno jest poprzedzony literaturowymi zastosowaniami sieci LSTM w układach FPGA nie związanymi z przykładem, lecz prowadzącymi do wskazania, że do implementacji tych sieci właściwa jest platforma FPGA. W dalszej części Autor opisuje proces kucia na zimno z dwoma uderzeniami stosowany do produkcji główek śrub. Do detekcji wadliwego uderzenia wykorzystywane są przebiegi z czujnika piezoelektrycznego mierzącego siły oddziałujące na obrabiany element o 256 próbkach każdy. Na podstawie przebiegów Autor zbudował klasyfikator w postaci sieci LSTM stwierdzający poprawne lub wadliwe uderzenie. Sieć składa się z jednej warstwy zawierającej 44 komórki LSTM i drugiej zbudowanej z pojedynczego neuronu z sigmoidalną funkcją aktywacji. Projekt sieci został szczegółowo opisany i zaimplementowany w strukturze układu FPGA. W implementacji sieci LSTM wykorzystano cztery warianty obliczania funkcji aktywacji. Zostały policzone: liczba wykorzystanych elementów LUT, przerzutników FF i procesorów DSP, a następnie wyznaczono liczbę taktów zegara, częstotliwość zegara, moc pobieraną ze źródła zasilania oraz maksymalną dokładność. W celach porównawczych dla tych czterech wariantów implementacji sieci LSTM w układach FPGA, zaprojektowano dwie wersje z procesorami CPU i dwie z procesorami GPU. Wszystkie wersje wykonane w układach FPGA mają znacząco mniejszy czas przetwarzania niż pozostałe. Jako wniosek Autor w podsumowaniu na stronie 117 podał, że najszybszy wariant 4 był 1760 razy szybszy od wariantu z procesorem GPU1 i 1829 razy szybszy od wariantu z procesorem GPU2. Wyjaśnił, że powodem był zbyt mały wymiar obliczeń, jak na kartę graficzną. Wyniki porównań czasu przetwarzania implementacji sieci LSTM w układach FPGA w stosunku do implementacji na procesorze z GPU przedstawione w literaturze [102], [103] i [105] są kilka razy mniejsze, a nie 1760 czy 1829 razy. Uzyskanie tak dobrych wyników powinno być uzasadnione np. oszacowaniem analitycznym. O ile w pracy na stronach 105÷110 wyjaśniono rzetelnie i szczegółowo jak zrealizowano projekt sieci LSTM implementowane na FPGA w układzie XCU250-FIGD2104-2L-E z płyty Alveo U250 Data Center Accelerator Card oraz programu Vivado, to o tyle warianty Wind+Keras+GPU1 i Ubun+Keras+GPU2 są przedstawione bez szczegółowych informacji implementacyjnych i można dociekać, że zostały zrealizowane w postaci niezoptymalizowanej. Gdyby napisać dedykowany kod na GPU tak jak powstał dedykowany kod dla FPGA wówczas taka implementacja byłaby szybsza i różnice w porównaniach znacznie mniejsze. Oznacza to porównywanie rozwiązania dedykowanego dla FPGA z nieoptymalnymi na innych platformach. Takie porównanie, może mieć głównie charakter poglądowy, a nie poznawczy. Porównania własnych osiągnięć powinny dotyczyć rozwiązań wykonywanych w identycznych lub zbliżonych warunkach, a najlepiej na tej samej platformie FPGA np. dla sieci rekurencyjnych GRU i zwykłych sieci rekurencyjnych, a także dla różnych systemów arytmetycznych. Porównanie mocy pobieranej ze źródła zasilania dla czterech wariantów implementowanych w układach FPGA pokazuje prawidłowy związek między liczbą użytych elementów, a mocą pobieraną ze źródła zasilania. Im więcej elementów, tym większa moc. Kolejne cztery implementacje sieci LSTM jako klasyfikatora kucia na zimno zostały opracowane na platformie ZYNQ. Różnią się one podziałem wykonywanych zadań w sieci pomiędzy programowaną logikę PL i system przetwarzania PS. Wśród tych rozwiań, trzy są szybsze od wariantów z procesorami GPU1 i GPU2, ale od dziesięciu nawet do ponad tysiąca razy wolniejsze od czterech wariantów implementowanych w układach FPGA. Wnioski podobnie jak we wcześniejszych badaniach prowadzą do stwierdzenia, że najlepsze rozwiązania realizacji sieci LSTM w zadaniach klasyfikacji w procesie kucia na zimno są implementowane w układach FPGA.

W rozdziale 4 Autor przedstawił własne rozwiązanie problemu uczenia sieci LSTM implementowanej w układach FPGA. Do uczenia sieci LSTM wykorzystał algorytm propagacji wstecznej najczęściej stosowany w sieciach rekurencyjnych. Analogicznie jak w rozdziale trzecim testy przeprowadził dla czterech struktur sieci różniących się metodą wyznaczania funkcji aktywacji. Określił ile elementów LUT, przerzutników FF oraz procesorów DSP zawiera każda ze struktur. W kolejnym etapie porównał ich czasy obliczeń z czasami obliczeń za pomocą dwóch struktur z procesorami CPU i dwóch z procesorami GPU, a także dwóch struktur uzyskanych za pomocą platformy ZYNQ. Czasy obliczeń za pomocą trzech wariantów sieci implementowanych w układach FPGA są ponad 45 razy krótsze niż dla implementacji z procesorami GPU, ponad 250 razy krótsze niż implementacja Python+CPU i implementacje wykonane na platformie ZYNQ. Te osiągnięcia także odbiegają od rozwiązań literaturowych wykorzystujących algorytm propagacji wstecznej opisanych w pracy J. Cong, Z. Fang, M. Lo, H. Wang, J. Xu and S. Zhang, "Understanding Performance Differences of FPGAs and GPUs," doi: 10.1109/FCCM.2018.00023. Analogicznie jak w rozdziale trzecim wnioski Autora prowadzą do stwierdzenia, że najlepsze rozwiązania układów uczenia sieci LSTM są implementowane w układach FPGA. W odniesieniu do porównań osiągnięć opisanych przez Autora można sformułować podobne zastrzeżenia jak w rozdziale trzecim.

W rozdziale piątym Autor podsumował swoje prace, wymienił główne osiągnięcia i wskazał na perspektywy dalszego rozwoju. W podsumowaniu Autor stwierdza, że tezy pracy zostały dowiedzione, a cele osiągnięte. Moim zdaniem treść pracy to potwierdza.

Wnioski końcowe

Praca jest napisana przejrzyście. Komórka LSTM, jej elementy oraz trzy rodzaje implementacji w układach FPGA: z wykorzystaniem algorytmu CORDIC, wielomianów Czebyszewa oraz zwykłych wielomianów zawierają szczegółowy opis parametrów modeli i sposobu analitycznego ich wyznaczania dla obliczeń numerycznych. Modele te zapewniają dużą dokładność obliczeń. Zaletą pracy jest także zaprojektowanie sieci LSTM oraz układu trenowania sieci LSTM, implementacja w układach FPGA i pomiary uzyskanych parametrów. W pracy znajdują się analizy porównawcze obliczeń implementacji sieci LSTM wykonanych na innych platformach niż ukałdy FPGA. Swoje nieliczne uwagi dla poprawy zawartości merytorycznej rozprawy wyraziłem w trakcie opisu osiągnięć w poszczególnych jej rozdziałach.

Podsumowując stwierdzam, że przedstawione badania są bardzo obiecujące w kontekście potencjalnych zastosowań, a otrzymane wyniki pozwalają sądzić, że układy FPGA mogą konkurować z popularnymi obecnie podejściami do obliczeń związanych z rekurencyjnymi sieciami neuronowymi. Rozprawa stanowi oryginalne rozwiązanie zagadnienia naukowego z zakresu sztucznych sieci neuronowych oraz może inicjować inne prace tego typu.

Do osiągnięć Autora wnoszących wkład do nauki zaliczam:

- opracowanie metody implementacji rekurencyjnej sieci LSTM w reprogramowalnych układach FPGA,
- optymalizację funkcji aktywacji w komórkach LSTM realizowanej różnymi metodami aproksymacji,
- opracowanie i implementację układu uczącego sieć LSTM w układzie FPGA,
- weryfikację praktyczną opracowanych układów sieci LSTM w procesie kucia na zimno.

Autor rozwiązał postawione przed nim zadania i użył do tego celu właściwych metod. Treść rozprawy jest potwierdzeniem dużej wiedzy teoretycznej, a przede wszystkim praktycznej Autora, czego dowodem są m.in. szczegółowo opisane projekty układów realizujących sieć LSTM w układach FPGA oraz praktyczne zastosowania proponowanej sieci do klasyfikacji wyrobów w procesie kucia na zimno. Zaliczam pracę do bardzo dobrych.

O wysokim poziomie osiągnięć naukowych Autora świadczy także fakt, że wyniki badań zostały opublikowane w trzech samodzielnych artykułach w czasopiśmie „*Parallel Processing Letters*" i jednym współautorskim z promotorem w czasopiśmie „*IEEE Access*". Oba czasopisma znajdują się na liście ministerialnej.

Konkluzja

Biorąc pod uwagę pozytywną ocenę rozprawy doktorskiej, stwierdzam, że spełnia ona wymagania stawiane przez odnośne przepisy „Prawa o szkolnictwie wyższym i nauce" z dnia 20 lipca 2018 roku w dyscyplinie „Informatyka Techniczna i Telekomunikacja". Wnoszę o przyjęcie rozprawy doktorskiej Pana mgr inż. Grzegorza Rafała Deca i skierowanie jej do obrony.

KWawrz