dr hab. Norbert Jankowski, prof. UMK

Katedra Informatyki Stosowanej, Uniwersytet Mikołaja Kopernika

ul. Grudziądzka 5, 87–100 Toruń
norbert@umk.pl, tel: (0 56) 6113307

Recenzja

rozprawy doktorskiej

„Beyond Traditional Curriculum Learning: Shaping The Optimization Path To Find Better Minima, Faster"
mgr inż. Izabela Krysińska

Przedstawione badania dotyczą ważnych aspektów procesu uczenia sztucznych sieci neuronowych. Najbardziej typowe uczenie sieci neuronowych polega na losowej kolejności analizy par treningowych, a autorka rozprawy zaproponowała nowe metody, które szukają odpowiedzi na pytanie: jaka kolejność wpływa możliwie najkorzystniej na proces uczenia w kontekście jego jakości i czasu trwania? *Programowe uczenie* w niektórych przypadkach jest nawet konieczne — zdarza się na przykład, że część zbioru jest mocno oporna na proces uczenia i wtedy, chcąc ten opór przełamać, należy wyjść poza standardowe uczenie. Jednak zasadniczo poszukiwanie najkorzystniejszego procesu uczenia powinno doprowadzić do lepszego wymodelowania granic decyzyjnych.

Warto też zwrócić uwagę, że to zagadnienie to tak naprawdę *task scheduler* tylko dla uczenia sieci neuronowych — mamy więc do czynienia ze sztandarowym problemem informatyki.

W rozdziale drugim autorka bardzo ładnie opisała zagadnienie uczenia programowego. Opisano główne składowe uczenia programowego, czyli funkcje oceniające próbki i funkcje nadzorujące proces uczenia, czyli decydujące na jakich danych będzie przebiegać uczenie. Zaprezentowano szereg już powstałych funkcji oceniających, jak i sposobów nadzoru do porcji uczących. Należy

uznać, że to wprowadzenie w uczenie programowe jest ciekawe i starannie zaprezentowane dla czytelnika.

Rozdział trzeci poświęcony jest omówieniu zaproponowanej koncepcji funkcji oceniającej typowość próbek uczących. Zaproponowano trzy wersje funkcji oceniającej próbki na bazie analizy podobieństwa pomiędzy próbkami i grafu, w którym krawędzie są tam, gdzie ich punkty–wierzchołki są podobniejsze niż średnio.

Testy użyteczności proponowanych funkcji oceniających zostały zaplanowane właściwie. Może jednak warto by rozważyć udział w testach większych zbiorów treningowych, bo wybrane są dość małe (wręcz zbyt małe). Możliwe, że przydatność zaproponowanego planowania uczenia w przypadku większych zbiorów byłaby korzystniejsza.

Analiza statystyczna rezultatów testów użyteczności tych funkcji oceniających pokazała, że niestety nie poprawiają one jakości uczenia w porównaniu ze zwykłym uczeniem stochastycznym.

Uwagi/pytania do rozdziału 3:

str 24. Degree centrality to nic innego jak stopień wierzchołka z teorii grafów dla grafu $\langle D, E \rangle$.

str 24. Opis paragrafu "Degree centrality" jest niejasny. Wprowadzono definicję "Degree centrality" i nie wskazano jak ona wpływa na wybór punktów do uczenia. Napisano tylko "Thus, training the model on these instances allows to generalize ...", lecz co oznacza "these" w tym miejscu? Wydaje się, że nic nie definiuje, którymi wierzchołkami/punktami następuje uczenie i w których fazach uczenia.

str 25. Pewna niespójność wkradła się w definicji "Entropy centrality". Definiując $N(x_i)$ napisano o najbliższych sąsiadach lecz wzór definiuje $N(x_i)$ jako wszystkie wierzchołki incydentne w E. Niektóre punkty x_i będą miały wielu sąsiadów, a to trochę wymyka się spod określenia "najbliżsi".

str 25. W definicji $c^*(x_i) = kp(c_k, x_i)$ chyba wkradł się jakiś błąd? Wydaje się, że chodziło o $c^*(x_i) = \arg\max_k \ p(c_k, x_i)$?

str 25. $S_{ECCA(x_i)}$ osiąga maksimum, gdy wszyscy sąsiedzi x_i w E są z klasy y_i (wartość 1) a minimum, gdy wszyscy sąsiedzi są z innej klasy (-1).

str 25. Dlaczego we wzorze na $S_{ECDA(x_i)}$ użyto różnicy? Co by się stało, gdyby różnicę zastąpić sumą?

W rozdziale 4 została zaproponowana ciekawa metoda, bazująca na wcześniejszym pomyśle, wykorzystującym zmienność gradientu. Zaproponowane zostało obserwowanie zmienności gradientu liczonego względem wektora wejściowego (nie wag sieci) w szeregu punktów w czasie. Zmienność gradientu ma służyć kontroli losowania próbek do kolejnych epok uczenia — próbki dla dużych wartości gradientu mają być losowane częściej, a o niskiej wartości rzadziej. W ten sposób sprawiono, że uczenie sieci nie traci czasu na dobrze nauczone próbki lecz uczenie następuje na próbkach, które mogą coś wnieść do uczenia. Jednak wymusza to przeliczanie wspomnianych gradientów co epokę, aby można było kierunkować niezbędny rozkład losowania próbek do kolejnych etapów uczenia.

Analiza porównawcza pokazała, że zaproponowane uczenie pozwala uzyskać lepszą jakość klasyfikacji. Widać również, szczególnie dla MNIST, że już po początkowej fazie krzywa poprawności klasyfikacji zaproponowanej metody jest wyżej niż uczenia standardowego. Analiza statystyczna wskazuje na istotność różnic jakości uczenia.

Uwagi/pytania do rozdziału 4:

str. 34 Opis danych "Webimmunization...". Może warto się zastanowić, czy powinno się myśleć o próbkach "Neutral" i "Pro-vaccine" jako próbkach klas rozłącznych. Mocno wydaje się, że te klasy zdecydowanie mogą nachodzić na siebie, w przeciwieństwie do próbek klasy "Anti-vaccine". Źródło tego wydaje się mieć miejsce w warstwie naukowego rozumienia problemu (lub nienaukowego — Anti-vaccine), czyli nie tylko potencjalny brak kontekstu może stanowić problem rozróżnienia pomiędzy "Pro-vaccine" i "Neutral".

str. 35 Autorka napisała: "This dataset is noted for being highly imbalanced...". Jak bardzo były niezbalansowane te dane?

str. 45 W pracy czytamy: "The most pressing challenge is scalability; computing gradients for the entire dataset with a large sliding window that seems to affect performance gains for each epoch is infeasible for very large-scale problems."

Dla ciągu $a_1, a_2, \ldots, a_n$ wyznaczanie wariancji kroczącej o oknie K to wyznaczanie wariancji z okna $a_i, \ldots, a_{K-1}$:

$$v_i = \frac{1}{K-1} \sum_{j=i}^{i+K-1} (a_j - \bar{a}_i)^2,$$

gdzie $\bar{a}_i = \frac{1}{K} \sum_{j=i}^{i+K-1} a_j$. Wyznaczenie wariancji w pierwszym oknie $(a_1, \ldots, a_K)$ ma złożoność $O(K)$, ale każde kolejne już tylko $O(1)$. Oczywiście muszą być pamiętane wartości, które są w oknie (progresywnie można zapominać wartości z lewej strony okna).

Czyli, gdy będą wyliczane wariancje gradientów dla kolejnych epok, to złożoność tego procesu nie będzie zależna od K.

Rozdział 5 opisuje szereg badań poświęconych analizie współdziałania metod harmonogramowania z różnymi metodami doboru porcji uczących, a także ze standardowym obecnie uczeniem SDG. Autorka zaproponowała badanie współdziałania porządkowania próbek względem ich typowości i łączeniu tego z różnymi sposobami organizacji porcji uczących (per próbka, per batch i per mini batch z rosnącą ilością próbek). Zaproponowane metody były porównywane ze standardowym losowym mini batchowym uczeniem SGD.

Rezultaty są ewidentnie ciekawe, ale i nieco zagadkowe. Widzimy, że metody korzystające z typowości próbek wygrywają ze standardowym uczeniem częściej, gdy używają SDG per próbka a rzadziej, gdy per mini batch. Wydawałoby się, że różnice wcale nie powinny być tak duże.

Natomiast gdy przyjrzymy się porównaniu z "batch repetition" to widzimy znaczną poprawę względem "mini-batch SGD". No i to właśnie jest zagadkowe, ponieważ doszło do ciekawej inwersji. W kontekście faktów, to można zauważyć, że wzór na batchowe SGD ma $\frac{1}{b}$ i jeśli używa się tego samego współczynnika uczenia (w tym kontekście brak informacji) dla wersji SGD per próbka i "mini-batch SGD", to ta druga efektywnie musi charakteryzować się wolniejszym uczeniem, a w konsekwencji niedouczeniem. Tę obserwację potwierdzałoby to, że gdy popatrzymy na wersję uczenia "batch repetition", to mamy do czynienia z j-krotnym powtórzeniem dla każdego batcha, a więc czynnik $\frac{1}{b}$ zamienia się w $\frac{j}{b}$ i mamy znacznie silniejsze uczenie (zakładając ten sam współczynnik uczenia).

Natomiast najciekawszy i najkorzystniejszy rezultat tego testu został uzyskany dla uczenia ze zmienną długością batcha. Ta wersja najczęściej wygrywała, choć nie najrzadziej przegrywała, ponieważ tutaj właśnie wersja z powtarzaniem batcha była najkorzystniejsza.

Zebranie powyższych obserwacji umożliwia stwierdzenie, że odpowiednio wczesne lepsze nauczenie modelu na próbkach typowych wspiera jakość uczenia. A to pokazuje, że zaproponowane zaplanowanie testu było słuszne, ciekawe i pożyteczne.

Drugie badanie dotyczyło zrozumienia kiedy i czy uda się otrzymać możliwie szybką zbieżność dla zadanej dokładności klasyfikacji. Zaplanowane zostały analizy czterech sposobów harmonogramowania (co było już treścią rozdziału 4) z trzema metodami zarządzania wielkością porcji uczących. Znów wyniki wcale nie były łatwo przewidywalne. Ewidentnie udało się odkryć, iż znacznie szybciej dochodzi do zbieżności przy wykorzystaniu metod VoG-ordered i VGL (tego można było się spodziewać) z dużymi batchami, a nie batchami o zmiennej ilości elementów (to

nie było oczywiste). Różnice są spore: najszybsza zbieżność to 32.2 dla VoG i 39.2 dla VGL epoki, a średnia zbieżność zajmuje 49.1 epok.

Można uznać to za zdecydowanie istotne dla planowania jak najciekawszych scenariuszy uczenia.

Jednak znaczącym dodatkowym atutem na rzecz modelu VGL jest to, co można zaobserwować na rysunku 5.2, a mianowicie stabilność procesu uczenia, którą tutaj warto porównać z VoG (w ogromnym kontraście). To jeszcze bardziej podnosi wartość znaczenia wiedzy o planowaniu procesu uczenia, ponieważ tak silna niestabilność może być ogromnym utrudnieniem w jakimkolwiek wdrożeniu.

Dodatkowo można zobaczyć, że VGL wraz z liniowym wzrostem porcji uczącej uzyskuje ewidentnie lepszą jakość klasyfikacji po tej samej ilości epok.

Uwagi/pytania do rozdziału 5:

str. 51 Punkt "Batch repetition". Czytamy "...the training process involves repeating each batch j times before moving to the next."

Jednak można zauważyć, że to jest niemal równoważne zwiększeniu współczynnika uczenia — j krotnie. Czyli de facto porównujemy uczenie na dwóch różnych "prędkościach", a nie znaczną zmianę strategii (rozmiar porcji uczącej jest wielokrotnie mniejszy od ilości próbek w zbiorze). Wydaje się, że bardziej interesujące mogłoby być zorganizowanie naboru próbek do kolejnych porcji uczących zgodnie z ich posortowaniem, ale z jednoczesnym uwzględnieniem balansu klas.

str. 52 Czytamy: "This initial objective, defined by the low-variance gradients from typical data, can be considered the "smoothed" function, J_0."

Na początku mamy mnóstwo losowych wartości w wagach sieci, a to nie jest zbyt gładki krajobraz. Dodatkowo nawet dla oczywistych (super-prototypowych) wektorów błędy są często duże i gradienty nie mogą być małe, bo jeszcze niczego się nie nauczyliśmy.

Czytamy: "The transition from one batch (B_t) to the next (B_{t+1}) represents a discrete step in the continuation schedule. When the training process moves from the first batch to the second, the underlying objective function being optimized is implicitly changed."

Gdyby dodatkowo pilnować proporcji klas, to prawdopodobnie można by uzyskać taką stabilność.

Rozdział 6 prezentuje ciekawą autorską metodę (TOL), którą jest uogólniony mechanizm uczenia z szeregiem pod-zadań (pod-celów). Zaprezentowano ciekawy pomysł jednoczesnego ujęcia, w całościowej funkcji celu, ważonej kombinacji podcelów uczenia. Natomiast to, że wagami kombinacji można zarządzać daje w rezultacie bardzo ciekawy model kontroli procesu uczenia (całości modelu).

Zaproponowany w ten sposób mechanizm użyto do zadań rozpoznawania mowy. Analizowane były zbiory (rozpoznawanie fonemów i słów) TIMIT, LnNor i Vibravox.

Konkretna realizacja TOC to włączenie w proces uczenia rozpoznawania ciągów fonemów i rozpoznawania ciągów liter (słów), co skutecznie doprowadziło do obniżenia błędów uczenia od 9.5% do 14%. Proponowana koncepcja pozytywnie wpłynęła również na szybkość zbieżności.

Zaproponowana koncepcja okazała się zdecydowanie pomyślna.

Podsumowanie

Jako główny dorobek doktorantki należy uznać rozprawę doktorską i artykuły naukowe z czasopism i konferencji. Autorka doktoratu wykazała się dużą wiedzą z zakresu sztucznych sieci neuronowych. Pani Krysińska zaproponowała nowe metody planowania (czy programowania) uczenia sieci neuronowych, a także zbadała jak najlepiej planować proces uczenia, wykorzystując do tego szereg różnych elementów wspomagających to planowanie. Warto zwrócić uwagę na staranność w zaplanowaniu poszczególnych badań — były to ciekawe scenariusze i prowadziły do weryfikacji celów. Należy zdecydowanie uznać, iż jest to istotny wkład naukowy w rozwój sieci neuronowych.

Dlatego można powiedzieć, że jest to istotny wkład w rozwój informatyki.

Doktorantka wykazała się bardzo dobrą wiedzą z informatyki. Umie się nią posługiwać a także widać sporą samodzielność.

Na pochwałę zasługuje jakość merytoryczna pracy, ale i jakość techniczna, dbałość o plan pracy, wzory, rysunki czy tabele.

Pani Izabela Krysińska jest autorką 6 publikacji, w tym 3 z obecnej listy punktowanych czasopism i konferencji.

Przedstawiona bibliografia jest obszerna i jak najbardziej właściwa. Praca została zaplanowano z dużą dbałością, jest jasna, a także charakteryzuje się pożądaną poprawnością naukowo–

techniczną.

Kończąc, oceniam pozytywnie dorobek doktorantki.

Uważam, że zaprezentowana rozprawa spełnia warunki dotyczące prac doktorskich i stawiam wniosek o dopuszczenie jej autorki do dalszych etapów przewodu doktorskiego.

Toruń, 11.01.2026

Norbert Jankowski