
Dr hab. inż. Wojciech Kotłowski, prof. PP
Instytut Informatyki (Institute of Computing Science)
Politechnika Poznańska (Poznań University of Technology)
ul. Piotrowo 2, 60-965 Poznań
tel: (+48) 61 665 2936
wkotlowski@cs.put.poznan.pl



Poznań, dn. 30 sierpnia 2022 r.

Recenzja rozprawy doktorskiej mgr. inż. Macieja Zamorskiego

*Representation learning on point cloud data with deep neural networks
(Uczenie się reprezentacji dla chmur punktów przy użyciu głębokich
sieci neuronowych)*

1 Wprowadzenie

Rozprawa doktorska mgr. inż. Macieja Zamorskiego dotyczy automatycznego przetwarzania danych trójwymiarowych, reprezentowanych jako chmury punktów, przy pomocy metod uczenia maszynowego z wykorzystaniem głębokich sieci neuronowych. W ostatnich latach jesteśmy świadkami ogromnych sukcesów głębokich sieci neuronowych w wielu problemach uczenia się z danych, między innymi w wizji komputerowej, przetwarzaniu języka naturalnego, rozpoznawaniu mowy i przetwarzaniu sygnałów, itp. O ile uczenie głębokie danych obrazowych jest już dojrzałą i dobrze eksplorowaną dziedziną z licznymi narzędziami ogólnego użytku rozwiązującymi w satysfakcjonującym stopniu podstawowe zadania rozpoznawania obrazów, o tyle temat przetwarzania danych trójwymiarowych nie doczekał się jeszcze należytej uwagi w badaniach naukowych. Z drugiej strony, rośnie liczba zastosowań, w których dane trójwymiarowe naturalnie się pojawiają. Stąd temat pracy doktoranta wydaje się świetnie trafiać we współczesne trendy badawcze i może prowadzić do istotnych potencjalnych zastosowań w nauce i przemyśle.

Z uwagi na wspomniane sukcesy sieci głębokich w analizie obrazów, ich użycie dla danych trójwymiarowych wydaje się być naturalnym i trafnym pomysłem. Nie jest to jednak zadanie proste, gdyż pojawiają się tutaj trudności z uwagi na specyficzne reprezentacje takich danych. Oczywiście, można potraktować trójwymiarowy obraz jako naturalne uogólnienie obrazu dwuwymiarowego, poprzez zastąpienie pikseli ich trójwymiarowymi odpowiednikami – woksalami – i np. wykorzystać warstwy konwolucyjne dodając kolejny

WPLYNĘŁO

30-08-2022

1

KDN-117/224/2022

wymiar do filtrów. Niestety, taka reprezentacja wydaje się w wielu przypadkach niepraktyczna z uwagi na liczbę wokseli budujących trójwymiarowe kształty, nawet przy stosunkowo niewielkiej rozdzielczości. Dodatkowo, woksele budujące jednostajną siatkę mogą być nieodpowiednie dla nierównomiernie ustrukturalizowanych danych lub danych z dużą ilością szczegółów. Stąd popularnym podejściem do reprezentacji danych trójwymiarowych są chmury punktów, czyli zbiory punktów definiujące dany kształt. Z jednej strony, ponieważ mamy do czynienia ze zbiorem, nie jest istotna kolejność, w której punkty są wyspecyfikowane, przez co metody analizy danych nie powinny zależeć od konkretnej permutacji listy punktów; z drugiej strony, istotna jest wzajemna relacja między punktami, ponieważ trzeba je potraktować całościowo jako opisujące jeden kształt. Te problemy powodują, że użycie modeli głębokich do analizy danych trójwymiarowych w reprezentacji w postaci chmury punktów jest z pewnością zadaniem wystarczająco ambitnym, aby stało się tematem rozprawy doktorskiej.

Autor pracy stawia sobie szereg celów i hipotez badawczych, dotyczących w ogólności problemu uczenia się reprezentacji chmur punktów. Są to problemy dotyczące mierzenia jakości reprezentacji, modelowania generatywnego (umożliwiającego generowanie chmur punktów z „rzeczywistego” rozkładu danych) przy użyciu odpowiedniej reprezentacji, samego tworzenia zwartych reprezentacji chmur punktów, w tym reprezentacji w postaci cech binarnych, grupowania (uczenia nienadzorowanego) chmur punktów, generowania chmur punktów o dowolnym rozmiarze, tzw. ciągłego uczenia się (ang. *continual learning*) nowych zadań dotyczących chmur punktów używając wiedzy z zadań poprzednich bez ich zapomnienia, uczenia się reprezentacji do dopełniania niekompletnych chmur punktów oraz analizy wyboru funkcji agregacji cech po zbiorze punktów i jej wpływu na jakość reprezentacji i trafność klasyfikacji. Robi wrażenie już sama mnogość celów badawczych, z jakimi zdecydował się zmierzyć Doktorant. W mojej opinii cele te są bardzo dobrze uzasadnione, interesujące, o dużym wymiarze praktycznym, a przede wszystkim nowatorskie i ambitne. Każdy z tych celów doprowadził do oryginalnych i istotnych wyników badawczych opublikowanych na czołowych konferencjach i czasopismach dziedziny.

2 Ocena struktury i zawartości pracy

Recenzowana rozprawa jest napisana czytelnie bardzo dobrym językiem angielskim. Liczy 141 stron i składa się z siedmiu rozdziałów. Całościowo jej strukturę oceniam całkowicie pozytywnie, a sam podział na rozdziały wydaje się zupełnie naturalny i zgodny z podziałem tematycznym treści (rozdziały dotyczące ściśle powiązanych ze sobą problemów badawczych i ich rozwiązań). Poniżej omówię zawartość poszczególnych rozdziałów, równocześnie starając się dokonać ich merytorycznej oceny.

Rozdział pierwszy ma charakter wprowadzenia. Nakreślona w nim została motywacja podjęcia badań w tematyce pracy, związana z rosnącą popularnością danych trójwymiarowych w zastosowaniach, a równocześnie stosunkowo niewielką liczbą publikacji i rozwiązań w tym zakresie. Następnie, w kolejnych sekcjach, opisane zostały cele badawcze Doktoranta wraz z przedstawieniem zaproponowanych rozwiązań i wyników eksperymentalnych. Każda z sekcji wprowadza krótko w konkretny problem badawczy, odpowiednio go motywując. Następnie przedstawia przegląd istniejących w literaturze rozwiązań, a tak-

że – w kontekście tych rozwiązań – opisuje zaproponowane przez p. Zamorskiego metody i osiągnięte rezultaty, odsyłając po szczegóły do konkretnego rozdziału pracy. Na koniec przedstawiony został ramowy plan rozprawy. Układ rozdziału przyjmuję bez zastrzeżeń, jako dobrze spełniający rolę wprowadzającą.

Rozdział drugi wprowadza czytelnika w rozważaną w pracy problematykę modelowania chmur punktów. Ma również charakter wprowadzenia i stanowi przegląd pojęć, problemów, metod i terminologii dotyczących tematu rozprawy. Krótko opisany zostaje problem uczenia się z danych, z rozdzieleniem na problemy uczenia nadzorowanego (*supervised learning*) i nienadzorowanego (*unsupervised learning*). Pojawia się notacja matematyczna, która będzie jednolicie używana do końca pracy. Wyjaśnione zostają też takie terminy jak redukcja wymiarowości (*dimensionality reduction*), uczenie się reprezentacji (*representation learning*), uczenie samo-nadzorowane (*self-supervised learning*) czy ciągłe uczenie się (*continual learning*). Większość z tych pojęć będzie przydatna przy zrozumieniu dalszych etapów pracy. Autor opisuje również dwa podstawowe podejścia do problemu statystycznego uczenia się: dyskryminatywne i generatywne, a także dokonuje przeglądu współcześnie używanych podejść do uczenia nienadzorowanego. Następnie przedstawiony został problem uczenia się reprezentacji. Myślę, że tutaj Autor świetnie poradził sobie z uchwyceniem samej definicji reprezentacji oraz wyszczególnieniem cech i własności, jakie reprezentacja powinna posiadać aby była użyteczna. W szczególności zwrócił uwagę na rozdzielność pojęć uczenia się reprezentacji i redukcji wymiarowości danych, zaletę z uchwycenia przez reprezentację poszczególnych ukrytych czynników charakteryzujących dane, konieczność abstrakcji reprezentacji od szczegółów problemu (dzięki czemu może być użyta w innych zadaniach o podobnej charakterystyce) czy zwiększenie gęstości informacji w stosunku do oryginalnej, wysokowymiarowej, ale często „pustej” przestrzeni. Następnie opisał metody uczenia się reprezentacji dla obrazów, rozpoczynając od podstawowych metod jak np. PCA, poprzez nowoczesne metody redukcji wymiarów typu *tSNE*, kończąc na metodach uczenia się reprezentacji w postaci uczenia kontrastowego (*contrastive learning*) czy autoenkoderów (*autoencoders*), w tym autoenkoderów wariacyjnych (*VAE*). Opis tych metod jest na tyle staranny, że powinien pozwolić je zrozumieć nawet czytelnikowi wcześniej ich nie znającemu (w moim przypadku dotyczyło to metodyki *contrastive learning*).

W dalszej części rozdziału opisane zostały metody reprezentacji danych trójwymiarowych (wielo-widokowy, oparty na wkselach, chmurach punktów, oraz kształcie w postaci punktów, krawędzi i ścian), każdorazowo wraz z wyczerpującym przeglądem literatury i opisem istniejących już metod analizy i uczenia się z danych. Szczegółowo przedstawiona została między innymi metoda *PointNet*, na której bazuje wiele późniejszych metod wprowadzonych przez Doktoranta. Po opisie metod reprezentacji pojawia się jeszcze opis zbiorów danych, wraz z przedstawieniem i uzasadnieniem ich selekcji w obrębie rozprawy, a następnie opis metryk używanych w celu uczenia i ewaluacji algorytmów, związanych z klasyfikacją, wyszukiwaniem informacji, dopasowaniem rozkładu prawdopodobieństwa, mierzeniem odległości między zbiorami, czy mierzeniem jakości generowanych danych (w modelach generatywnych). Podsumowując, rozdział jest bardzo obszerny, ale dzięki temu znakomicie spełnia swoją rolę jako wprowadzenia w tematykę pracy, przeglądu metod i szczegółowego przeglądu istniejących prac. Prawie wszystkie pojęcia wprowadzone w tym rozdziale są później konsekwentnie używane w konstrukcji i analizie eksperymentalnej własnych algorytmów.

Rozdział trzeci dotyczy uczenia się reprezentacji dla modelowania generatywnego. Rozpoczyna się od krótkiego opisu stanu badań w tym zakresie, a następnie formalnie przedstawia rozważany problem. Następnie pojawia się praca własna Autora, metoda *3-d Adversarial Autoencoder (3dAAE)*. Metoda ta służy do generowania chmur punktów przez głęboką sieć typu *autoencoder*, dodatkowo wyposażoną w mechanizm uczenia współzawodniczącego (*adversarial learning*), podobny do tego zastosowanego w metodach typu *GAN*. Architektura jest oparta na istniejącej sieci *PointNet*, z uwagi na jej niezależności od permutacji danych wejściowych (punktów) i współdzielenie wag w obrębie każdego z punktów wraz z funkcją agregującą cechy ukryte do wektora cech podsumowującego całą chmurę. Architektura została wzbogacona o mechanizm uczenia się rozkładów normalnych, podobny do *VAE*, bazujący na zwracaniu parametrów średniej i odchylenia, a następnie generowania danych w oparciu o reparametryzację, dzięki czemu możliwa jest wsteczna propagacja gradientu. Regularyzacja reprezentacji względem rozkładu a priori została tutaj uzyskana właśnie dzięki mechanizmowi *adversarial learning*, poprzez wprowadzenie generatora tworzącego chmury punktów z wektora reprezentacji, wraz z dyskryminatorem próbującym rozróżnić wygenerowane w ten sposób chmury od chmur uzyskanych z rozkładu a priori. Autor zdecydował się tu na zastąpienie klasycznej funkcji celu *GAN* funkcją opartą o odległość Wassersteina z uwagi na jej stabilność i unikanie zjawiska *mode collapse*. Autor świetnie łączy tutaj istniejące metody w celu uzyskania finalnego algorytmu bardzo dobrze radzącego sobie z postawionym problemem.

W dalszej części rozdziału przedstawiona została modyfikacja wprowadzonego algorytmu, w której celem jest utworzenie reprezentacji składającej się z wektorów binarnych (metoda *3dAAE-Beta*). P. Zamorski zrezygnował tutaj z rozkładu normalnego, zastępując go rozkładem beta z parametrami gwarantującymi z dużym prawdopodobieństwem uzyskanie wartości cech reprezentacji bliskimi wartościom 0 lub 1. Równocześnie, część kodująca (*encoder*) została pozbawiona (o ile dobrze rozumiem) elementu losowania wektorów reprezentacji. Tutaj miałem pewien problem ze zrozumieniem dokładnego mechanizmu działania algorytmu (w szczególności, czy coś pojawia się w miejsce „triku reparametryzacji”) i nie jestem pewien, co wymusza na enkoderze tworzenie reprezentacji binarnej (lub bliskiej jej) – czy jest to jedynie efekt współzawodniczenia z dyskryminatorem? Tym niemniej, jak pokazuje późniejszy eksperyment, udało się osiągnąć postawiony cel i metoda tworzy binarne reprezentacje o bardzo dobrych własnościach dopasowania do rozkładu danych.

W kolejnym podrozdziale pojawia się podejście *3dAAE-C*, które jest modyfikacją metody *3dAAE* w celu grupowania chmur punktów. Autor uzyskał je poprzez wymuszenie reprezentacji w postaci wektorów typu *one-hot encoding*, wskazujących tylko na jedną grupę. W samym algorytmie enkoder wytwarza „miękką” wersję wektorów *one-hot* poprzez użycie warstwy *soft-max* na wyjściu, ale współzawodniczy (między innymi) z dyskryminatorem, który swoje cechy uzyskuje generując wektory *one-hot* z jednostajnego rozkładu po kategoriach (grupach). Podejście wydaje się bardzo pomysłowe, a późniejszy eksperyment pokazuje, że świetnie radzi sobie w praktyce. Ciekawi mnie jednak, dlaczego potrzebny był Autorowi dodatkowy dyskryminator i reprezentacja wytwarzana przez enkodera z normalnym rozkładem wektorów reprezentacji (czy metoda działała bez tego elementu gorzej?). Nie jestem również pewien, dlaczego dyskryminator nie miałby w procesie uczenia wykorzystywać faktu, że dane z rozkładu a priori do niego trafiające mają postać „twardych” wektorów *one-hot*, w odróżnieniu od „miękkich” wektorów (uzyskanych z warstwy *soft-*

max) pochodzących od encodera, dzięki czemu zawsze byłby w stanie odróżnić oba typy danych?

P. Zamorski przedstawia następnie metodę *HyperCloud*, która – w odróżnieniu od metod poprzednich – jest w stanie generować chmury punktów o dowolnej liczności, zamiast odgórnie ustalonej i niejako „zaszytej” w architekturze sieci. Odbywa się to poprzez utworzenie generatora, która zamiast chmury punktów zwraca na wyjściu parametry (wagi) kolejnej sieci, mapującej punkty z jednostkowej sfery na odpowiednie punkty pokrywające reprezentowany przez chmurę kształt. Pomysł jest zaskakujący, ale okazuje się bardzo trafny i prowadzi do metody z sukcesem generującej chmury punktów o dowolnym rozmiarze. Tutaj zastanawia mnie użycie sfery jako zbioru wejściowego, a dokładniej sposób, w jaki tego typu odwzorowanie mogłoby poradzić sobie z chmurami pokrywającymi całą objętość danego kształtu?

W dalszej części rozdziału przedstawiono jeszcze jedną metodę, *Conditional Invertible Flow (CIF)*, znacznie różniącą się pod względem działania od poprzednich, bazującą na odwzorowaniu chmury punktów, jak również samych punktów, na wektory pochodzące z rozkładu normalnego. Następnie warunkując na wektorze reprezentacji chmury i wygenerowaniu losowych wektorów normalnych, jednostki implementujące przepływ zostają odwrócone w celu otrzymania danych mających charakterystykę podobną do chmury na wejściu. Metoda ta bardzo dobrze wykorzystuje ideę odwracalnych przepływów, dodając warunkowanie na wektorze reprezentacji chmury do generatora pojedynczych punktów, dzięki czemu przy generacji brane są pod uwagę własności całego kształtu, który chmura reprezentuje. Metoda opisana jest stosunkowo zwięźle, przez co jej zrozumienie było trudniejsze niż metod opisanych w poprzednich podrozdziałach. Tym niemniej pomysł uważam za bardzo interesujący, a jego pozytywna weryfikacja przedstawiona jest w eksperymencie obliczeniowym.

Pod koniec rozdziału pojawia się wspomniany już obszerny eksperyment, zawierający zarówno wszystkie metody opisane w rozdziale, jak również metody porównawcze (*1-GAN* oraz *AE*) z wcześniejszych prac. Autor zbadał metody pod różnymi kątami, przyglądając się ich możliwością rekonstrukcji kształtu (weryfikującej problem nadmiernego dopasowania się do danych), jak i dopasowaniem rozkładu generowanych danych (przy użyciu trzech metod pomiaru dopasowania, bazujących na dwóch różnych metrykach odległości zbiorów), a także trafności klasyfikacji przy użyciu liniowego klasyfikatora typu *SVM* bazującego na wektorach reprezentacji. Eksperyment jednoznacznie pokazał, że zaproponowane metody osiągają bardzo dobre wyniki, przewyższające lub konkurencyjne dla wyników metod istniejących. Zweryfikowano również jakość reprezentacji binarnej (w zadaniu klasyfikacji oraz wizualnie), empirycznie dowodząc jej użyteczność. Interesującym dodatkiem było przedstawienie ilustracyjnych przykładów generowanych chmur jako obrysów kształtów rzeczywistych przedmiotów, analiza pokrycia przestrzeni zmiennych ukrytych (wraz z interpolacją między wektorami reprezentacji i ich wpływem na wynikową chmurę punktów), a także weryfikacja abstrakcji reprezentacji i rozdzielania przez nią czynników stojących za danymi. Wyniki te były dla mnie bardzo interesujące, szczególnie, że Autor powiązał je na estetycznych rysunkach ze zmianami kształtu rzeczywistych przedmiotów. Równocześnie doceniam wysokie wartości miar jakości uzyskiwanych przez wprowadzone metod, potwierdzające ich użyteczność dla rozwiązywania rzeczywistych problemów.

Kolejne rozdziały są mniej obszerne, niż omówiony powyżej rozdział trzeci. W rozdziale czwartym przedstawiono problem ciągłego uczenia się (*continual learning*) reprezentacji dla chmur punktów. Celem jest tutaj nauczenie się kolejnych zadań, bazując na wiedzy uzyskanej z uczenia się zadań poprzednich. Równocześnie, metoda nie powinna „zapominać” zadań poprzednich i również na nich utrzymać wysoką trafność predykcji. Doktorant zaproponował tutaj metodę, która podsumowuje dane związane z zadaniem już nauczone do informatywnego podzbioru o mniejszym rozmiarze, który jest później używany podczas uczenia zadań kolejnych, aby „przypominać” algorytmowi o zadaniach już nauczonych (metoda *Random Compression Rehearsal*). Metoda czerpie z pomysłów z poprzednich rozdziałów, używając podobnej architektury bazującej na *PointNet*, przy czym uczenie odbywa się poprzez jednoczesną optymalizację generatora danych oraz klasyfikatora. Dla każdego nowego zadania, rozszerzane jest następnie wyjście klasyfikatora o kolejny wymiar. RCR porównane jest w eksperymencie obliczeniowym na czterech zbiorach danych zarówno z podejściami bazowymi, określanymi jako *baseline* (uczenie się nowych zadań bez pamiętania o starych) oraz *toplines* (uczenie się wszystkich zadań jednocześnie), jak również innymi istniejącymi metodami ciągłego uczenia się. Wyniki jednoznacznie wskazują, że RCR osiąga najlepszą trafność klasyfikacji na wszystkich zbiorach danych. Co jest dla mnie zaskakujące, RCR jest również lepsze od wspomnianych *toplines*.

Kolejny rozdział dotyczy problemu uzupełniania niekompletnych chmur punktów. W tym celu Autor zmodyfikował bazową metodę AAE, rozszerzając ją o dodatkową sieć, której przekazywana jest po otrzymaniu reprezentacji ponownie całościowa informacja o chmurze punktów, pozwalająca lepiej modelować jej kształt. Podejście to zostało nazwane *Double PointNet*. W eksperymencie obliczeniowym porównano działanie bazowych metod bez dodatkowej sieci (*AE*, *VAE*, *AAN-N*, *AAE-B* – dwie ostatnie są metodami używanymi zasadniczo w rozdziale trzecim, tworzącymi reprezentacje w postaci wektorów o rozkładzie normalnym oraz wektorów binarnych), z analogicznymi metodami wyposażonymi w dodatkową sieć. Te drugie okazały się w każdym przypadku znacząco lepsze. Pomysł świetnie sprawdził się więc przy dopełnianiu chmury punktów. Interesuje mnie, czy wprowadzenie tej dodatkowej sieci nie polepsza również metod w *ogólnym* przypadku, np. gdy celem jest generacja chmur punktów lub ich klasyfikacja.

Rozdział szósty dotyczy klasyfikacji chmur punktów i skupia się w głównej mierze na funkcji agregacji używanej przez sieć *PointNet* (*max pooling*), aby przejść z macierzy wektorów reprezentacji dla poszczególnych punktów do całościowego wektora reprezentacji dla chmury. Doktorant zaproponował tutaj inne funkcje agregacji, rozpoczynając od prostych funkcji opartych na pojedynczych statystykach niezmienniczych ze względu na permutację (takich jak suma, średnia, mediana i maksimum), poprzez użycie wielu statystyk równocześnie, kończąc na wielowartościowych statystykach danych typu *top-k*. Następnie, powyższe funkcje agregacji zostały porównane w eksperymencie obliczeniowym na reprezentacjach o różnym wymiarze, za pomocą trafności klasyfikacji i zrównoważonej trafności klasyfikacji (*balanced accuracy*). Doktorantowi udało się znaleźć funkcje agregacji zachowujące się lepiej od bazowego *max pooling* (w zależności od typu miary jakości, są to wielowymiarowe statystyki lub średnia). Zmierzona została również różnorodność punktów, na których opiera się całościowo funkcja tworząca globalną reprezentację (zastosowanie w przypadku funkcji maksimum i mediany). Powyższe wyniki są nowatorskie i bardzo interesujące, i pozwalają w stosunkowo łatwy sposób (poprzez podmianę funkcji agregacji) polepszyć

wyniki istniejących metod klasyfikacji dla chmur punktów.

Ostatni z rozdziałów zawiera podsumowanie pracy.

3 Ocena wkładu oryginalnego

Rozprawa jest oparta na kilku artykułach, których współautorem jest Doktorant:

- Zamorski, M., Stypułkowski, M., Karanowski, K., , Trzciński, T., Zięba, M. (2022). Continual learning on 3D point clouds with random compressed rehearsal. *arXiv pre-print*.
- Stypułkowski, M., Kania, K., Zamorski, M., Zięba, M., Trzciński, T., Chorowski, J. (2021). Representing point clouds with generative conditional invertible flow networks. *Pattern Recognition Letters*.
- Spurek, P., Winczowski, S., Tabor, J., Zamorski, M., Zięba, M., Trzciński, T. (2020). Hypernetwork approach to generating point clouds. *International Conference on Machine Learning (ICML)*.
- Zamorski, M., Zięba, M., Świątek, J. (2020). Generative Modeling in Application to Point Cloud Completion. *International Conference on Artificial Intelligence and Soft Computing (ICAISC)*.
- Zamorski, M., Zięba, M., Świątek, J. (2020). Comparison of Aggregation Functions for 3D Point Clouds Classification. *Intelligent Information and Database Systems (IIDS)*.
- Zamorski, M., Zięba, M., Klukowski, P., Nowak, R., Kurach, K., Stokowiec, W., Trzciński, T. (2020). Adversarial Autoencoders for Compact Representations of 3D Point Clouds. *Computer Vision and Image Understanding (CVIU)*.
- Stypułkowski, M., Zamorski, M., Zięba, M., Chorowski, J. (2019). Conditional Invertible Flow for Point Cloud Generation. *NeurIPS 2019 Workshop on Sets and Partitions*.
- Zamorski, M., Zdobylak, A., Zięba, M., Świątek, J. (2019). Generative Adversarial Networks: recent developments. *International Conference on Artificial Intelligence and Soft Computing (ICAISC)*.
- Zamorski, M., Zięba, M. (2019). Semi-supervised learning with Bidirectional GANs. *Intelligent Information and Database Systems (IIDS)*.

Dorobek uznaję za bardzo obszerny, z nawiązką spełniający wymogi do finalizacji przewodu doktorskiego. Potwierdza to artykuł opublikowany na konferencji *International Conference on Machine Learning (ICML)*, będącej czołową konferencją w dziedzinie o bardzo selektywnym procesie recenzyjnym, a także artykuł w prestiżowym czasopiśmie *Pattern Recognition Letters (impact factor 3.756)*.

Rozprawa zawiera wiele oryginalnych wyników, które omówiłem już w poprzedniej części recenzji, w związku z tym tutaj jedynie je podsumuję:

- Modelowanie generatywne chmur punktów poprzez użycie nowatorskiego pomysłu opartego na metodyce *adversarial autoencoders*, łączące w sobie użycie sieci *PointNet*, autoenkoderów wariacyjnych oraz sieci *GAN*.
- Uzyskanie zwartych reprezentacji binarnych chmur punktów poprzez użycie rozkładu beta w wyżej opisanej architekturze.
- Grupowanie chmur punktów poprzez użycie rozkładu na kategoriach wraz z uczeniem się reprezentacji i zaprzęgnięciem metod uczenia współzawodniczącego.
- Generacja chmur punktów o dowolnym rozmiarze dzięki architekturze uczącej się odwzorowania sfery jednostkowej na kształt opisywany przez chmurę punktów (*HyperCloud*).
- Ciągłe uczenie się zapobiegające zapominaniu poprzednich zadań dzięki kompresji danych dotyczących poprzednich zadań i wykorzystywaniu ich w dalszym procesie uczenia.
- Dopełnianie niekompletnych chmur punktów za pomocą algorytmu *Double PointNet*, zawierającego dodatkową sieć określającą globalne cechy reprezentacji na podstawie wektorów reprezentacji kształtu wraz z wektorami lokalizacji punktów.
- Analiza postaci funkcji agregacji i jej wpływu na jakość reprezentacji i trafność klasyfikacji.

Krótko podsumowując, uznaję, że wymienione na wstępie pracy cele udało się Doktorantowi w pełni osiągnąć.

4 Uwagi dyskusyjne

Nie kwestionując wartości całościowych wyników zawartych w rozprawie, chciałbym zgłosić poniżej kilka uwag, głównie w formie pytań bądź dyskusji.

- Współzmienniczość / niezmienniczość względem rotacji i translacji. Wydaje się, że chmury punktów opisujące kształty powinny być klasyfikowane / generowane w ten sam (lub współzmienniczy) sposób po zastosowaniu całościowej rotacji chmury bądź jej translacji w dowolnym kierunku. Zastanawia mnie czy bazowa metoda tworzenia reprezentacji, używana przez Doktoranta, faktycznie to zapewnia. Jeśli tak nie jest, czy Doktorant widzi sposób, aby wziąć pod uwagę taką wiedzę dziedzinową poprzez odpowiednią modyfikację architektury sieci?
- Reprezentacja binarna i typu *one-hot*. W rozdziale trzecim Doktorant używa rozkładu beta na wejściu dyskryminatora do tworzenia reprezentacji binarnej (podrozdział 3.3.2), natomiast w następnej metodzie (dotyczącej grupowania, podrozdział 3.3.3) rozkład jest podmieniany na rozkład na kategoriach. Widzę tutaj pewną różnicę w podejściu do problemów. Jeśli oprzeć się na rozkładzie na kategoriach z 3.3.3, dlaczego nie tworzyć reprezentacji binarnej w 3.3.2 zastępując rozkład beta (który i tak

skoncentrowany jest wokół wartości 0 i 1) rozkładem na dwóch kategoriach (tzw. rozkładem dwupunktowym)? Z drugiej strony, jeśli rozkład beta miałby w 3.3.2 być bardziej użyteczny od dwupunktowego, dlaczego w takim razie nie zastąpić rozkładu na wielu kategoriach (w metodzie grupowania w 3.3.3) analogicznym do rozkładu beta rozkładem Dirichleta (również odpowiednio skoncentrowanym na wartościach skrajnych)?

- *Reparameterization trick*. Czy dobrze rozumiem, że reparametryzacja (a tym samym losowanie punktów w warstwie reprezentacji) nie jest częścią implementacji w metodach opisanych w sekcjach 3.3.2 i 3.3.3 (binarne cechy oraz grupowanie). Czy Doktorant widzi możliwość wprowadzenia dodatkowej losowości (np. poprzez podejście *Gumbel-max*) i czy byłoby to użyteczne dla wyników metod?
- Metoda *HyperCloud*. Nie jestem pewien, czy dobrze uchwyciłem ideę metody *HyperCloud*. Czy wynikowa sieć T , której wag uczy się *AAE*, przetwarza punkt po punkcie, niezależnie od innych punktów? jeśli tak, to jaką mamy gwarancję, że istnieje *jakikolwiek* (ciągłe) odwzorowanie z jednostkowej sfery w kształt opisywany przez chmurę punktów?
- Wyniki *toplines*. W rozdziale 4, w tabeli 4.1, zaproponowana metoda *RGR* wydaje się osiągać wyniki lepsze od metody *Supervised*, która uczy się wszystkich zadań na raz i powinna być nieosiągalnym celem. Czy Doktorant potrafi wyjaśnić, skąd bierze się ten efekt?

5 Konkluzja końcowa

Rozprawę oceniam bardzo dobrze, co w powyższej recenzji podkreśliłem wielokrotnie. Problemy badawcze, z którymi zmierzył się Doktorant, są ambitne i istotne dla postępu w dziedzinie uczenia się reprezentacji dla chmur punktów. Sama rozprawa charakteryzuje się bardzo wysokim poziomem merytorycznym i zawiera wiele interesujące rezultatów, jednoznacznie potwierdzających użyteczność i praktyczność zaproponowanych algorytmów. Jestem również pod wrażeniem, jak wiele różnych wątków i metod udało się Doktorantowi uwzględnić w obrębie pracy. W mojej opinii Doktorant wykazał się znakomitymi umiejętnościami prowadzenia badań naukowych. Wszystkie postawione w rozprawie cele zostały osiągnięte.

W związku z tym **rozprawę oceniam jako spełniającą wymogi stawiane pracom doktorskim i wnoszę o dopuszczenie mgr. inż. Macieja Zamorskiego do dalszych etapów przewodu doktorskiego, równocześnie sugerując wyróżnienie pracy.**



dr hab. inż. Wojciech Kotłowski

