PREZENTACJA

• ADAM – Slajd Tytułowy`  
  
Dzień dobry – prezentację chcielibyśmy zacząć od podziękowań za naukę podczas tegorocznych studiów podpylomowych w tych ciężkich, zdalnych czasach. Tematem naszego projektu jest szeroko rozumiane pojęcie text-to-image generation. W obecnych czasach znaleźć możemy różnorodne sposoby podjęcia tego tematu – od GANów, poprzez transformery i inne rozwiązania state-of-the-art. W naszej prezentacj postaramy się przybliżyć najważniejsze elementy testowanego przez nas rozwiązania, a także wyniki eksperymentów, który przy użyciu tego rozwiązania przeprowadziliśmy.

• ADAM – Dall-e

Główną motywacją do podjęcia tego tematu było opublikowanie przez firmę OpenAI w 2021 publikacji dotyczącej modelu Dall-e czyli ich największego modelu pozwalającego na generowanie obrazów na podstawie tekstu. Autorzy opisują ten model jako model o 12-miliardach parametrów, który potrafi nie tylko generować obrazy przedstawiające rzeczywiste obiekty, ale też tworzyć abstrakcyjne twory, które mogą łączyć w sobie wiele elementów jak widoczne na przykładzie krzesło w kształcie awokado. W dużym skrócie – model dall-e składa się z 2 elementów – części generatywnej, która nie została na ten moment upubliczniona, a także części odpowiedzialnej za powiązanie tekstu z obrazem nazwanej przez autorów CLIP. Ta część modelu została upubliczniona i odniesiemy się do niej w dalszej części prezentacji.

• MACIEK – GANy  
  
Esencją naszej pracy jest zastosowanie modeli generatywnych do tworzenia obrazów z tekstu, zatem w naszej pracy opisujemy i wykorzystujemy architekturę generative adversarial networks, popularnie znane pod akronimem GAN. W pracy opisujemy jak skonstruowane są tego typu sieci, jak są uczone i ewaluowane. Do rozwiązania które opiszemy w dalszej części prezentacji użyliśmy szeroko znanych i przetrenowanych modeli typu GAN, dokładniej StyleGAN2 oraz BigGAN. Jak możemy przeczytać na slajdzie, obie architektury są relatywnie nowe, aczkolwiek powstały poprzez stopniowe polepszanie poprzednich architektur, np. StyleGAN czy Progan, zatem powstały poprzez lata uświetniania poprzednich dokonań w tej dziedzinie i poprzednich architektur. Oparliśmy prace o te dwa modele, ponieważ ich części generatywne są dostępne jako open-source, a to dla nas istotne, gdyż właśnie te generatywne części są wykorzystywane w rozwiązaniu. (BigGAN na przykład nie ma ogólnodostępnej części dyskryminatora). Drugim i w zasadzie najistotniejszym czynnikiem jest dosyć dużą różnica tych dwóch modeli pod względem sposobu nauki oraz potencjału generatynego. StyleGAN2 jest dostępny jako wyspecjalizowane modele służące do generowania konkretnych obiektów (samochody, koty) podczas gdy BigGAN jest ogólnym modelem do różnych fraz. Będziemy o tym wspominać jeszcze w dalszej części pracy.

• ADAM – CLIP

Jak wspominaliśmy wcześniej jednym z elementów tworzących model DALL-E jest CLIP czyli Contrastive Language-Image Pre-Training. Jest to sieć neuronowa wyuczona na ponad 400 milionach par obraz-podpis, które zostały pobrane z internetu przez autorów. Służy on do zambedowania obrazu oraz podpisu w taki sposób by otrzymane wektory przechowywały jak najwięcej informacji z obu elementów. W naszym rozwiązaniu wykorzystujemy propowany w CLIPie sposób porównywania embeddingów wygenerowanego przez model zdjęcia, wraz z zaembedowanym podanym przez użytkownika zapytaniem przy użyciu podobieństwa cosinusowego.   
Na zdjęciu widzimy przykład użycia CLIPa jako algorytmu klasyfikacyjnego – podając mu zdjęcie oraz listę potencjalnych podpisów możemy znaleźć najlepszy z nich.

• MACIEK – Algorytmy ewolucyjne

Więc z jednej strony mamy GAN, który bierze pewien n-wymiarowy wektor i na jego podstawie generuje obraz. Mamy również tekst podany przez użytkownika. Zatem pytanie brzmi, w jaki sposób nakierować wektor wchodzący do generatora, aby tworzył on obraz zbieżny z tekstem podanym przez użytkownika. I tutaj wykorzystujemy algorytmy ewolucyjne wraz z opisanym przez Adama modelem CLIP. Algorytmy ewolucyjne korzystają z pewnej funkcji, nazywanej fitness function, czyli pewna funkcja straty, która w naszym przypadku pochodzi z modelu CLIP, jest to odwrotność podobieństwa cosinusowemu miedzy embeddingiem tekstu i embeddingiem obrazu wygenerowanego przez model.

Algorytmy ewolucyjne są można powiedzieć brakującym elementem układanki która właśnie opisujemy. Z jednej strony mamy GANy jako model generujący obrazy, mamy również model CLIP który określa dokładność (bliskość) wygenerowanego obrazu oraz pewnego tekstu. Algorytmy ewolucyjne są w rozwiązaniu wykorzystywane do naprowadzenia generatora do stworzenia obrazów które będą jak najbliżej wejściowej frazy. np. generator modelu StyleGAN2 wykorzystuje wektor o długości 512 elementów jako wsad do sieci. Można rozumieć ten wektor jako input, który definiuje wygenerowany obraz. Zadaniem algorytmu jest aby zoptymalizować wartości wektora tak, aby generowały obraz jak najbliższy wejściowej frazie. Za moment pojawi się w prezentacji graf opisujący cały przepływ informacji w naszym rozwiązaniu. Wykorzystaliśmy dwa popularne modele ewolucyjne, podstawowy algorytm genetyczny oraz tak zwaną ewolucję różnicująca. Główny powód tych wyborów to pewna prostota tych algorytmów i implementacji, jednocześnie chcieliśmy wypróbować dwa podejścia w celu porównania. Ewolucja roznicujaca jest często określana jako lepszy (szybciej ubiegający algorytm), niemniej często kosztem trudniejszej kalibracji. Pokażemy dalej jakie wyniki udało nam się uzyskać korzystając z obu algorytmów.

• MACIEK – Framework – opis

I tutaj mamy wspomniany przez nas wcześniej graf opisujący działanie całego mechanizmu. Jest to swego rodzaju konsolidacja wszystkich elementów o których wspominaliśmy. Myślę, że warto zacząć od lewej strony oraz od góry, gdyż są to można powiedzieć startowe punkty całej maszynerii. Podajemy frazę, na przykład ‚czerwony samochód’, która jest enkodowxana w przestrzeni modelu CLIP. Jednocześnie do modelu, jakim jest generator np. BigGAN, podawany jest losowy szum, tzn. zbiór wektorów (tyle, ile wielkośc populacji) o długości 128 elementów. Następnie za pomocą tych wektorów model generuje obrazy, które są porównywane z wejściową zakodowana frazą. Majac ewaluacje każdego z osobników dokonujemy kroku algorytmu ewolucyjnego, w którym dokonuje się mieszanie i mutacja osobników, tym samym tworząc silniejsze potomstwo i pozostawiając lepsza populacje. Ten proces zapętla się ponownie z generowaniem obrazów i finalnie, po przejściu określonej liczby kroków algorytmu, wybieramy najlepiej oceniony obraz z populacji. Na koniec, po dokonaniu większej liczby takich eksperymentów dokonujemy ewaluacji, o której będziemy niebawem mówić.

…

• ADAM – Framework – przykład  
  
Kiedy omówiliśmy już nasz framework – przeanalizujmy prosty przykład wykonania naszego algorytmu. Zaczynamy od konfiguracji algorytmu – w tym przypadku postanowiliśmy wybrać algorytm StyleGAN2 w wersji dla generowania twarzy, dodatkowo algorytm GA, oczywiście algorytm CLIP i ustalamy 200 iteracji algorytmu.  
Zadanym przez nas sformułowaniem jest „a blond girl with a smile”. Na 1 zdjęciu możemy zobaczyć batch z modelu, który otrzymaliśmy po 100 epoce – widzimy już, że przy tak prostym przykładzie dla „StyleGAN2” większość zdjęć przypomina bez problemu blondwłosą, uśmiechniętą kobietę. Zdjęcie ostateczne oddaje nam najlepsze zdjęcie z algorytmu genetycznego na podstawie metryki jaką jest cos-sim.

• MACIEK – GA vs DE

W pracy opisujemy porównanie pomiędzy zastosowaniem obu algorytmów ewolucyjnych wspomnianych wcześniej. Głównym wnioskiem jest to, ze algorytm genetyczny wydal się nam nieco lepszy. Uśrednione wyniki kilkuset eksperymentów wskazały, ze szybciej docierał on do bardzo dobrych rezultatów. Niemniej, przy wystarczająco długiej optymalizacji wyniki obu algorytmów były mocno porównywalne. Tutaj mamy przykład obrazków wygenerowanych z frazy ‚żółty samochód w mieście’ przy użyciu obu algorytmów, bardzo podobne wyniki naocznie, niemniej metryka wskazywała lepsze wyniki dla GA. Tutaj jeszcze taka ciekawostka, można zauważyć pewna interesującą kwestie na tych dwóch zdjęciach, mianowicie swego rodzaju ‚przeciek’ koloru z samochodu na pozostała część obrazu, Tzn. Normalnie miasto nie byłoby aż tak żółte. Jest to zjawisko które zaobserwowaliśmy kilkakrotnie w różnych scenariuszach, zwłaszcza przy korzystaniu z żywych kolorów. Mianowicie cały embedding modelu CLIP ma pewien problem w zrozumieniu sensu przekazywanej frazy i nie interpretuje żółtego samochodu jako związku wyrazów i nie zapewnia semantycznego powiązania tylko na poziomie tych dwóch wyrazów. Zamiast tego, po prostu stara się, aby obraz był jak najbardziej zolty. JEst to problem który zapewne wpada w nieco inne zagadnienie, może trochę bardziej z zakresu NLP i architektur typu transformer, niemniej wydało nam się dosyć ciekawe.

• MACIEK – StyleGAN vs BigGAN

Kolejnym porównaniem jest opisanie różnic miedzy modelami BigGAN i StyleGAN2 przy generowaniu różnych obrazów. Tutaj zauważyliśmy dosyć wyraźną zależność - największe znaczenie przy generowaniu różnych obrazów ma to, na jakim datagecie dany model był trenowany. W przypadku modeli StyleGAN2, generują one obrazy bardzo wysokiej jakości ale w bardzo wąskiej dziedzinie, tzn. Modele są nauczone jedynie to tworzenia obiektów jednej klasy, jak samochód, koty czy konie. W przypadku modelu BigGAN, te modele generują znacznie bardzo różnorodne obrazy. Tutaj widzimy dwa przykłady, pierwszy to czerwony gotycki kościół, który wygenerowaliśmy za pomocą dedykowanego modelu StyleGAN2-church, oraz to jak z ta fraza poradził sobie biggan. Widzimy niesamowicie dobra jakoś w pierwszym przypadku i w zasadzie trudny do opisania obiekt w drugim, czerwony kolor i coś na wzór okiennic gotyckich.

Natomiast gdy rozpatrzymy drugi przypadek, czyli rowerzystę klauna na księżycu, to zauważymy, ze jedynie biggan stara się chociaż trochę wygenerować obiekt który nosi namiastkę podanej frazy. Model stylegan w zasadzie wylosował obrazki, natomiast biggan chociaż trochę dopasował do frazy, z przynajmniej zdjęcia są ukierunkowane w pewien styl, co świadczy o pewnym rozumowaniu sieci.

Możemy powiedzieć, ze ścierają się tutaj dwa podstawowe aspekty ganow - różnorodność oraz jakość. Notabene, w tym celu tez została stworzona metryka Inception Score do porównywania mocy generatywnej ganom, oparta o dywergencje Kullbacka-Leiblera, która tez opisujemy w pracy.

• ADAM – Ewaluacja  
  
Pomijając pojedyncze eksperymenty, które miały nam odpowiedzieć na postawione przez nas pytania, postanowiliśmy również sprawdzić jak testowane przez nas rozwiązanie sprawdzi się w szerszej skali. W tym celu postanowiliśmy przeprowadzić ewaluację, którą zasugerował nam paper dotyczący modelu CLIP. Mianowicie: dla dwóch popularnych datasetów – CIFAR10 oraz ImageNet znaleźliśmy wyuczone na nich klasyfikatory o jak największej skuteczności, następnie wygenerowaliśmy dla 10 wybranych klas z obu datasetów po 512 zdjęć, podaliśmy je do klasyfikatorów i obliczyliśmy jaki % wygenerowanych przez nas zdjęć potrafił oszukać klasyfikatory by uznały nasze zdjęcia za podobne do tych ze zbioru treningowego. Wyniki prezentują się następująco:  
- Pierwszy dataset – CIFAR10 wybraliśmy ze względu na małą liczbę klas, którą posiada – 10. Otrzymany średni % accuracy na wygenerowanych przez nas zdjęciach nie przekroczył 25. Analizując pokazywaną macierz błędów doszliśmy do wniosków, że zwierzęta były często mylone ze sobą (zagęszczenie na środku), a maszyny ze sobą (airplane, automobile).   
- Drugim datasetem był ImageNet – z jednej strony w porównaniu do CIFAR10 zawiera on zdecydyowanie więcej klas – ponad 2000, z drugiej strony, model BigGAN, który wykorzystujemy był na nim uczony, więc cięzko było powiedzieć, czy wyniki będą prezentować się lepiej czy gorzej niż dla CIFARA. Są one następujące – nie przekroczyliśmy 23% skuteczności, ale zważywszy na poziom skomplikowania zadania, odbieramy ten wynik pozytywnie – co piąte zdjęcie wygenerowane przez nas na podstawie nazwy klasy pozwalało oszukać klasyfikator. Przykłady dla wszystkich klas, które wygenerowaliśmy w obu datasetach ewaluacyjnych można znaleźć w załączniku do naszej pracy – tutaj przedstawiamy przykładowy z nich czyli 5 zdjęć, które oszukały klasyfikator dla klasy LLAMA i 5 które nie dały rady go oszukać.

• MACIEK – Wnioski

•

Prawdopodobnie dobra kontynuacja tej pracy byłoby wypróbowanie innych modeli i innych aklgorytmow do nawigacji przestrzeni generatywnej sieci. Także moglibyśmy poczynić dokładniejsze badania nad różnicami algorytmów, ich zupełnie inna kalibracja mogłaby potencjalnie doprowadzić do innych rezultatów, których tutaj już nie zbadaliśmy. Także chcielibyśmy się dużo bardziej zgłębić w temat semantyki i zbadać różne zachowania modelu dla różnych, bardziej skomplikowanych fraz. Tak jak na przykład przeciek koloru wydał nam się bardzo ciekawy, być może byłyby tez inne wnioski. W niedalekiej przyszłości planujemy przeprowadzenie dodatkowych eksperymentów i opublikowanie ich w formie artykułu na platformie medium.

Z uwag już bardziej personalnych, być może mniej do samej pracy - wydaje nam się, ze dużo się nauczyliśmy, zarówno w trakcie studiów jak i w trakcie pisania pracy. Często rozwiązania oparte o NLP i CV przeważają, zarówno w naukowych pracach jak i praktycznych zastosowaniach, niemniej temat samej architektury typu GAN wydal nam się od początku bardzo ciekawy i cieszymy się, ze udało nam się zgłębić go trochę bardziej. Mamy nadzieje, ze nasza praca wydala się być ciekawa i dziękujemy!

UWAGI Z CALLA:

Siema, w prezentacji omówimy temat text-to-image generation…,  
- co omówimy  
- motywacja/geneza

• DALL-E: ADAM

- OpenAI wypuscili GPT3+CLIP, tylko CLIP jest udostepniony, jest to jedno z wielu podejść bla bla, zero-shot classifier, dużo parametrów, state-of-the-art

• GANy (po 2-3 kropki – kto stworzyl, do czego sluzyl, na czym wyuczone) MACIEK  
- zastępuje czesc generacyjna w Dall-e, my uzywamy przykladow state-of-the-art. – StyleGAN2 i BigGAN – stylegan skupia się na konkretnych srodowiskach, Biggan lepiej generalizuje z zalozenia

• CLIP (ten z pracki) (punkty) ADAM

• Algorytmy ewolucyjne (po 2 kropki) MACIEK  
-

• Framework MACIEK