

KNUM x GOLEM Hackathon 2022

Machine Learning: Few-Shot Image Classification

Zadanie powstało dzięki zaangażowaniu firmy nomagic.ai

Problem

Problematykę treningu z kilku przykładów treningowych nazywamy **few-shot learningiem**.

Zadanie polega na stworzeniu klasyfikatora obiektów z danych pozyskanych z kilkudziesięciu przykładów referencyjnych, a następnie klasyfikację obiektów, które znajdują się w skrzynce. Każdy przykład referencyjny zawiera zdjęcie skrzynki z przedmiotami **jednego typu** oraz odpowiednimi bounding boxami umożliwiającymi wycięcie danych. Każdy przykład walidacyjny/ewaluacyjny zawiera zdjęcie skrzynki, w której mogą znajdować się przedmioty **wielu typów** wraz z odpowiednimi bounding boxami. Celem zadania jest przydzielenie dla każdego bounding boxa ze zbioru walidacyjnego(ewaluacyjnego) odpowiedniej klasy obiektu, korzystając z modelu wytrenowanego na danych referencyjnych tylko part1 (folder: **reference_images_part1**, plik z adnotacjami: **reference_images_part1.json**). Dopuszczane jest skorzystanie z pretrenowanych checkpointów dostępnych w internecie, lecz zakazane jest dotrenowywanie modelu na czymkolwiek innym niż danych pozyskanych z **reference_images_part1**.

Rozwiązanie będzie ewaluowane na dwóch rodzajach danych. W pierwszym wariancie należy wykorzystać zdjęcia referencyjne **reference_images_part1** (wraz z odpowiadającymi adnotacjami **reference_images_part1.json**) oraz model wytrenowany na nich w celu klasyfikacji każdego bounding boxa z każdego zdjęcia z **images_part1_test** (wraz z odpowiadającymi adnotacjami **images_part1_test_public.json**).

W drugim wariancie należy wykorzystać zdjęcia referencyjne **reference_images_part2** (wraz z odpowiadającymi adnotacjami **reference_images_part2.json**) oraz **ten sam model co wyżej - czyli taki, który podczas treningu (lub finetuningu z gotowego checkpointa) widział tylko reference_images_part1** - w celu klasyfikacji każdego bounding boxa z każdego zdjęcia z **images_part2_test** (wraz z odpowiadającymi adnotacjami **images_part2_test_public.json**).

Uwaga 1 - dane do ewaluacji mają taki sam format jak dane do walidacji. Zostaną one dostarczone godzinę przed końcem hackathonu.

Uwaga 2 - jeśli wśród danych referencyjnych brakuje, któregoś typu przedmiotu, który znajduje się w jsonie - należy nie brać pod uwagę tej klasy podczas treningu i inferencji. Żadne id nie ulegają przez to zmianie.

Dane można obejrzeć przy użyciu skryptu **public_dataset/view.py**, np.

Wyświetlenie danych referencyjnych part1:

```
python3 public_dataset/view.py --data-path
public_dataset/reference_images_part1 --labels-path
public_dataset/reference_images_part1.json
```

Wyświetlenie danych walidacyjnych:

```
python3 public_dataset/view.py --data-path  
public_dataset/images_part1_valid --labels-path  
public_dataset/images_part1_valid.json
```

Cel Zadania

Rozwiązanie można testować na części walidacyjnej. Jest ona w takim samym formacie jak część ewaluacyjna. W pliku `images_part1_valid.json` zawarte są słowniki, wśród nich takie o kluczach:

- `images` - tutaj zawarte są meta informacje o zdjęciach.
- `annotations` - każde entry zawiera **pojedynczy** bounding box zdjęcia o id `image_id`, przedstawiający obiekt o kategorii `category_id`. W entry znaleźć można również dodatkowe dane takie jak to, czy obiekt był przystąpięty na zdjęciu.
- `categories` - globalne mapowanie id kategorii (w tym entry `id` odpowiadające `category_id` z adnotacji) na nazwę obiektu `name`.

Publiczne części testowego datasetu (`images_part1_test_public.json` oraz `images_part2_test_public.json`) mają ukryte `category_id`. Celem zadania jest podmienienie wartości `null` na id kategorii, w taki sposób jak to wygląda w części walidacyjnej.

Ostateczny wynik to procent poprawnie zaklasyfikowanych bounding boxów w `images_part1_test_public.json` oraz `images_part2_test_public.json`.

Rozwiązaniem zadania są pliki `images_part1_test_solution.json` oraz `images_part2_test_solution.json`, powstałe z `images_part1_test_public.json` oraz `images_part2_test_public.json`, kod i link do checkpointa, który powstał w wyniku tego kodu. Istotną częścią rozwiązania jest zapewnienie reprodukowalności treningu (np. poprzez ustawienie wszystkich seedów w używanych frameworkach).

Rozwiązanie należy przestać na adres quiz@mlinpl.org przed końcem hackathonu z nazwą zespołu w tytule.