

Systemy sztucznej inteligencji

dokumentacja projektu DigitRecognizer

Jambor Daniel
Grupa 2D

Kozieł Wojtek
Grupa 2D

Matula Kamil
Grupa 2D

10 maja 2020

Część I

Opis programu «JAK DLA MNIE TO STYKNIE - ILE MOŻNA PISAĆ O TAK PROSTEJ RZECZY»

Program DigitRecognizer służy do rozpoznawania ręcznie napisanych działań i wyświetlanie ich wyniku. Użytkownik pisze na specjalnym polu liczby całkowite oraz jeden z czterech zaimplementowanych wyrażeń arytmetycznych (dodawanie, odejmowanie, dzielenie i mnożenie), a program wyświetla końcowy wynik podanego wcześniej działania. Program korzysta z bazy danych 'MNIST', która składa się łącznie z 70 000 ręcznie napisanych cyfr oraz z autorskiej bazy danych oznaczeń matematycznych.

Instrukcja obsługi «MOŻNA COŚ DOPISAĆ»

Aby uruchomić program, należy uruchomić plik *DigitRecognizer.exe*. Do prawidłowego działania aplikacji wymagany jest plik *weights.txt*, który znajduje się w tym samym folderze co plik *DigitRecognizer.exe*. Jeśli plik nie istnieje, należy pobrać program jeszcze raz. Po uruchomieniu programu, użytkownik zobaczy:

- Białe pole, po którym użytkownik może pisać równania.
- Pole tekstowe, gdzie będą wyświetlane wszystkie komunikaty odnośnie działania programu.
- Przycisk 'Uruchom sieć neuronową' - jest on jedynym dostępnym przyciskiem po uruchomieniu programu - tworzy on sieć neuronową i wczytuje wcześniej wyuczone wagi.
- Przycisk 'Wyczyść' - przycisk który czyści zawartość białego pola.
- Przycisk 'Oblicz' - przycisk który pobiera narysowane przez użytkownika działanie i oblicza wynik.

Aby dostać wynik, użytkownik musi narysować na białym polu równanie. W razie pomyłki może nacisnąć przycisk 'Wyczyść', aby zacząć jeszcze raz. Po narysowaniu równania wystarczy wcisnąć przycisk 'Oblicz', po czym w polu tekstowym pojawi się wynik równania.

Dodatkowe informacje «NIE MAM POJĘCIA CO TUTAJ MOŻNA DAĆ ;-;»

Wymagania itd.

Część II

Opis działania

Tutaj uwzględniamy część matematyczną. Opisujemy całą teorię np.: dla zadania związanego z sieciami neuronowymi - opisujemy całą budowę, algorytm uczenia i wszystkie wzory. Dla zadania związanego z kombinatoryką opisujemy całą teorię kombinatoryczną potrzebną do zrozumienia zadania (mile widziany przykład obliczeniowy).

Algorytm

Tutaj opisujemy rozwiązanie zadania. Dla przedmiotu programowanie będzie to wykorzystanie matematyki z poprzedniego zadania itd. Dla SSI będzie to ogólne działanie przetwarzania danych w oparciu o modele matematyczne z poprzedniego zadania.

Pseudokod tworzymy w L^AT_EX. Przykład:

Data: Dane wejściowe liczba k

Result: Brak

$i := 0;$

while $i < k$ **do**

 Drukuj na ekran liczbę i ;

if $i \% 2 == 0$ **then**

 | Wydrukuj informację, że liczba i jest liczbą parzystą;

else

 | Wydrukuj informację, że liczba i nie jest liczbą parzystą;

end

end

Algorithm 1: Algorytm drukowania informacji o liczbie parzystej/nieparzystej.

Bazy danych

Nauka sieci neuronowej wykorzystywała dwie bazy danych:

MNIST

THE MNIST DATABASE of handwritten digits - baza danych składająca się z 60 000 próbek treningowych oraz 10 000 próbek walidacyjnych. Dane są przechowywane w formacie *.idx3-ubyte* (w przypadku samych obrazków) oraz *.idx1-ubyte* (w przypadku etykiet). Pliki składają się z wartości typu *ubyte* i 32 bit integer. W przypadku plików przechowujących grafiki, pierwsze cztery wartości przechowują:

- 'Magic number' - liczba kontrolna
- Liczba zdjęć
- Szerokość jednego zdjęcia
- Wysokość jednego zdjęcia

- Kolejne pixele zdjęć. Pixele te przyjmują wartości od 0 (które reprezentuje kolor biały) do 255 (reprezentacja koloru czarnego)

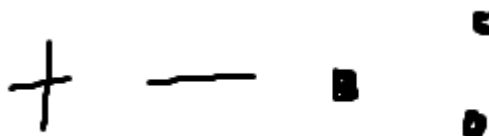
Pliki przechowujące etykiety zbudowane są na podobnej zasadzie:

- 'Magic number' - liczba kontrolna
- Liczba etykiet
- Kolejne etykiety

Wczytywanie bazy danych polega na wczytaniu pierwszych czterech wartości z pliku zawierającego zdjęcia oraz pierwszych dwóch z pliku zawierającego etykiety. Następnie do tablicy dwuwymiarowej wczytywane jest 784 pixeli (28 x 28), a do drugiego wymiaru odpowiadająca zdjęciu etykieta (dla jedynki będzie to 1 itp.). Takie dane są już gotowe do przesłania ich do sieci, jednak etykiety zostały przebudowane na potrzeby projektu.

Baza oznaczeń matematycznych

Baza oznaczeń matematycznych została zrobiona na potrzeby projektu. Baza opiera się na plikach .png, które zawierają zestawy każdego znaku:



Jeden plik zawiera 50 takich zestawów umieszczonych w jednym wierszu. Łącznie baza danych składa się z 600 znaków, z czego 10% jest traktowana jako część walidacyjna. Za pomocą napisanego algorytmu, plik jest dzielony na 200 osobnych obrazków, których pixele są zapisywane w tablicy tak samo jak w przypadku bazy danych MNIST. Z racji tego, że możliwe są tylko cztery etykiety, dodawane one są naprzemiennie, gdyż kolejność oznaczeń jest taka sama (plus, minus, mnożenie, dzielenie).

Implementacja

Opis, zasada i działanie programu ze względu na podział na pliki, następnie funkcje programu wraz ze szczegółowym opisem działania (np.: formie pseudokodu, czy odniesienia do równania)

```
1  Tutaj wklejamy fragment kodu, który chcemy opisac
2  (bez polskich znaków).
```

Testy

Tutaj powinna pojawić się analiza uzyskanych wyników oraz wykresy/pomiary.

Eksperymenty

Sekcję używamy gdy porównywaliśmy dwa lub więcej algorytmów, albo wykonywaliśmy jakies pomiary.

Warto dodać jakies wykresy jako obraz, albo tabele z wynikami.

Wszystkie wyniki powinny być opisane/poddane komentarzowi i poddane analizie statystycznej.

Pełen kod aplikacji

`1 Tutaj wklejamy pełen kod.`
