

Rozpoznawanie Podanych Liczb

Sara Fijołek i Miron Kwiatkowski

1 Wstęp

Celem aplikacji jest stworzenie modelu sieci neuronowej, który potrafi rozpoznać, czy podana liczba jest parzysta, oraz sprawdzić, czy jest liczba pierwsza. Aplikacja została zaimplementowana w języku Python przy użyciu bibliotek NumPy, TensorFlow, oraz Flask w celu umożliwienia interakcji z użytkownikiem poprzez stronę internetową.

2 Kod aplikacji

2.1 Generowanie danych i model sieci neuronowej

Poniżej przedstawiono kod odpowiedzialny za generowanie danych oraz stworzenie modelu sieci neuronowej:

Listing 1: Generowanie danych i model sieci neuronowej

```
1 import numpy as np
2 from tensorflow.keras.models import Sequential
3 from tensorflow.keras.layers import Dense
4
5 # Funkcja sprawdzająca, czy liczba jest pierwsza
6 def is_prime(n):
7     if n <= 1:
8         return False
9     for i in range(2, int(n ** 0.5) + 1):
10         if n % i == 0:
11             return False
12     return True
13
14 # Generowanie danych treningowych
15 def generate_data(limit=1000):
16     X = np.array(range(2, limit + 2)).reshape(-1, 1)
17     y_parity = np.array([1 if x % 2 == 0 else 0 for x in X.flatten()])
18     y_prime = np.array([1 if is_prime(x) else 0 for x in X.flatten()])
19     y = np.column_stack((y_parity, y_prime))
20     return X, y
21
22 # Tworzenie modelu
23 def create_model():
24     model = Sequential([
25         Dense(128, input_dim=1, activation='relu'),
26         Dense(64, activation='relu'),
27         Dense(32, activation='relu'),
28         Dense(2, activation='sigmoid')
29     ])
```

```

30     model.compile(optimizer='adam', loss='binary_crossentropy', metrics
31                 =['accuracy'])
32     return model

```

2.2 Serwer Flask

Kod serwera internetowego umożliwia użytkownikowi interakcje z modelem za pośrednictwem formularza na stronie internetowej:

Listing 2: Serwer Flask

```

1  from flask import Flask, render_template, request
2  import numpy as np
3
4  app = Flask(__name__)
5
6  @app.route('/', methods=['GET', 'POST'])
7  def index():
8      result = None
9      if request.method == 'POST':
10         try:
11             number = int(request.form['number'])
12             if number <= 0:
13                 raise ValueError("Liczba musi by dodatnia i
14                                     ca kowita")
15             except ValueError:
16                 result = {"error": "Podaj liczb  ca kowit  wi ksz  od
17                                     zera!"}
18             else:
19                 prediction = model.predict(np.array([[number]]))
20                 is_even = prediction[0][0] > 0.5
21                 is_prime = prediction[0][1] > 0.5
22                 result = {
23                     'number': number,
24                     'even': "parzysta" if is_even else "nieparzysta",
25                     'prime': "pierwsza" if is_prime else "z o ona"
26                 }
27             return render_template('index.html', result=result)
28
29 if __name__ == '__main__':
30     app.run(debug=True)

```

2.3 Kod HTML

Kod strony internetowej pozwala na wprowadzenie liczby i wyświetlenie wyniku:

Listing 3: Kod HTML

```

1  <!DOCTYPE html>
2  <html lang="pl">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale
6          =1.0">
7      <title>Sprawdzenie liczby</title>
8      <link rel="stylesheet" href="{ url_for('static', filename='styl.
9          css') }}">

```

```

8 </head>
9 <body>
10 <h2>Sprawd , czy liczba jest parzysta i czy jest liczb pierwsz </
    h2>
11
12 <form method="POST">
13 <label for="number">Prosz , podaj liczb :</label>
14 <input type="number" id="number" name="number" min="1" step="1"
    required>
15 <button type="submit">Sprawd wynik</button>
16 </form>
17
18 {% if result %}
19 {% if result.error %}
20 <p style="color: red;">{{ result.error }}</p>
21 {% else %}
22 <h3>Wyniki dla liczby {{ result.number }}:</h3>
23 <p>Parzysto : {{ result.even }}</p>
24 <p>Pierwszo : {{ result.prime }}</p>
25 {% endif %}
26 {% endif %}
27 </body>
28 </html>

```

2.4 Kod CSS

Stylizacja strony została zaimplementowana w pliku `styl.css`:

Listing 4: Kod CSS (styl.css)

```

1 body {
2     font-family: Arial, sans-serif;
3     background-color: #45f02e;
4     color: #333;
5     text-align: center;
6 }
7
8 h2 {
9     color: #0056b3;
10 }
11
12 form {
13     margin-top: 20px;
14 }
15
16 input, button {
17     padding: 10px;
18     font-size: 16px;
19     margin: 5px;
20 }
21
22 button {
23     background-color: #0056b3;
24     color: white;
25     border: none;
26     cursor: pointer;
27 }

```

```

28
29 button:hover {
30     background-color: #004494;
31 }

```

2.5 Kod JavaScript

Kod JavaScript odpowiada za walidację formularza i poprawne działanie interfejsu:

Listing 5: Kod JavaScript (script.js)

```

1 document.addEventListener('DOMContentLoaded', function() {
2     console.log("JavaScript zaadowany poprawnie!");
3
4     document.querySelector('form').addEventListener('submit', function(
5         event) {
6         const numberInput = document.getElementById('number');
7         const numberValue = parseFloat(numberInput.value);
8
9         // Sprawdzamy, czy liczba jest dodatnia i całkowita
10        if (numberValue <= 0 || !Number.isInteger(numberValue)) {
11            alert("Podaj liczbę całkowitą większą od zera!");
12            event.preventDefault(); // Zapobiega wysłaniu formularza
13        }
14    });
15 });

```

3 Wnioski

- 1. Poprawność działania: Model skutecznie klasyfikuje liczby jako parzyste/nieparzyste oraz pierwsze/złożone, wykorzystując dane treningowe wygenerowane dynamicznie.
- 2. Interfejs użytkownika: Zastosowanie Flask oraz HTML pozwala użytkownikowi wprowadzać liczby i natychmiast otrzymywać wyniki w przyjazny sposób.
- 3. Edukacyjna wartość projektu: Aplikacja uczy użytkownika podstaw wnioskowania maszynowego oraz integracji modeli uczenia maszynowego z aplikacjami internetowymi.

4 Zrzuty ekranu z aplikacji

Sprawdź, czy liczba jest parzysta i czy jest liczbą pierwszą

Proszę, podać liczbę:

Sprawdź wynik

Liczba: 2

Parzystość: parzysta

Pierwszość: pierwsza

Sprawdź, czy liczba jest parzysta i czy jest liczbą pierwszą

Proszę, podaj liczbę:

Sprawdź wynik

Wyniki dla liczby 1:

Parzystość: nieparzysta

Pierwszość: złożona

Sprawdź, czy liczba jest parzysta i czy jest liczbą pierwszą

Proszę, podaj liczbę:

Sprawdź wynik

Wyniki dla liczby 34:

Parzystość: parzysta

Pierwszość: złożona