

Algorytmy sztucznej inteligencji w Przemysle 4.0

”Identyfikacja i kategoryzacja emocji w dźwięku”

Gracjan Sadkowski 256100
Jan Dworczak 256232

Poniedziałek 13:15 – 15:00, semestr zimowy 23/24

1 Zakres projektu

1. **Przygotowanie danych** – zebranie nagrań dźwiękowych około 5 osób wyrażających różne emocje (np. radość, smutek, gniew, strach, zaskoczenie). Nagrania będą odpowiednio przygotowane do dalszej analizy pod względem długości i standaryzacji. Będą istniały różne typy wypowiedzianych słów. W razie potrzeby zastosowane będzie filtry dolnoprzepustowy, który tłumi sygnały o częstotliwościach wyższych od częstotliwości granicznej
2. **Wgranie danych** – skorzystanie z bibliotek w języku python do przetworzenia nagrań na dane numeryczne (np. tablice numpy).
3. **Ekstrakcja cech** – wydobycie z nagrań takich parametrów, jak tempo mowy, ton dźwięku, amplituda oraz częstotliwość fundamentalna i eksperymentowanie z różnymi zestawami cech lub technikami ekstrakcji cech.
4. **Przygotowanie danych do uczenia** – podzielenie danych na zbiór treningowy i testowy oraz przypisanie etykiet emocji do odpowiednich próbek.
5. **Wybór modelu uczenia maszynowego** – Wybór spośród algorytmów takich, jak SVM, sieci neuronowe, las losowy lub uczenie głębokie oraz optymalizacja hiperparametrów używanych w modelu.
6. **Wytrenowanie modelu**
7. **Przetestowanie modelu na danych testowych** – Skorzystanie z metryk ewaluacyjnych, takich jak dokładność, precyzja itd., aby ocenić wydajność modelu.

8. **Ocena rzeczywistej skuteczności algorytmu w praktycznych warunkach** – Po wytrenowaniu algorytmu na zbiorze treningowym i przetestowaniu na zbiorze testowym, badanie skuteczności algorytmu na nowych nagraniach wgranych do środowiska lub na żywo z dźwiękiem pobieranym przez mikrofon.

2 Wstęp Teoretyczny

Analiza emocji w dźwięku jest obszarem badań, który ma na celu rozpoznawanie emocji wyrażanych w nagraniach dźwiękowych. Celem tego kodu jest zbudowanie modelu uczenia maszynowego, który jest w stanie przewidywać emocje na podstawie cech ekstrahowanych z plików dźwiękowych.

Etap 3 - przygotowanie

W trzecim etapie projektu skupiono się na przygotowaniu środowiska pracy oraz zbioru uczącego dla badanego problemu, którym jest rozpoznawanie emocji. Projekt ten będzie realizowany w środowisku Python, a dane uczące będą zawierały cztery kategorie emocji: 'calm', 'angry', 'fearful', 'happy'.

Przygotowanie środowiska pracy

Do realizacji projektu wybrano środowisko Python ze względu na jego popularność, elastyczność i dostępność wielu bibliotek wspierających uczenie maszynowe. W ramach przygotowania środowiska pracy wykonano następujące kroki:

- Zainstalowano niezbędne biblioteki, takie jak `numpy`, `pandas`, `matplotlib`, oraz biblioteki dedykowane do uczenia maszynowego, takie jak `scikit-learn` i `tensorflow`.
- Skonfigurowano środowisko pracy w narzędziu `Jupyter Notebook`, umożliwiając interaktywny rozwój i eksperymentowanie z kodem.

Zbiór uczący

Dane uczące składają się z czterech kategorii emocji: 'calm', 'angry', 'fearful', 'happy'. Zbiór ten został pozyskany z wiarygodnych źródeł zawierających nagrania głosowe reprezentujące różne emocje. W celu utrzymania równowagi pomiędzy kategoriami, przeprowadzono odpowiednią analizę i przekształcenia danych.

Etap 4: Implementacja pierwszej metody

W tym etapie projektu skupiono się na implementacji pierwszej metody bazującej na sztucznej inteligencji w celu rozpoznawania emocji związanych z dźwiękiem. Poniżej znajduje się opis implementacji oraz kluczowe fragmenty kodu.

Implementacja metody

Do implementacji metody wykorzystano język programowania Python oraz popularne biblioteki takie jak `librosa`, `scikit-learn`, i `tensorflow`. Poniżej przedstawiamy kluczowe aspekty implementacji:

- **Wczytywanie danych:** Dane dźwiękowe zostały wczytane z folderu, gdzie każda emocja jest reprezentowana przez pliki dźwiękowe w formacie `.wav`. Funkcja `wczytaj_dane` odpowiada za ekstrakcję cech z plików dźwiękowych oraz przygotowanie danych do uczenia.
- **Model sieci neuronowej:** Zaimplementowano model sieci neuronowej przy użyciu biblioteki `tensorflow.keras`. Model ten składa się z warstw `Dense` z funkcją aktywacji `relu`, warstw `Dropout` w celu zapobiegania przeuczeniu, oraz warstwy wyjściowej z funkcją aktywacji `softmax`.
- **Trenowanie modelu:** Model został skompilowany z funkcją straty `categorical_crossentropy` i optymalizatorem `adam`. Następnie przeprowadzono trenowanie modelu na danych treningowych przez 100 epok.
- **Ocena modelu:** Po zakończeniu trenowania, oceniono skuteczność modelu na danych testowych, wyświetlając dokładność i stratę.
- **Przewidywanie emocji:** Zaimplementowano funkcję `przewidz_emocje`, która przyjmuje ścieżkę do nowego pliku dźwiękowego, normalizuje dane na podstawie wcześniej obliczonych średnich i odchyłeń standardowych, a następnie przewiduje emocje za pomocą wytrenowanego modelu.

Wizualizacja wyników

W celu wizualizacji wyników, przygotowano funkcje do generowania wykresów. Funkcja `historia_uczenia` prezentuje wykresy dokładności i straty w trakcie uczenia modelu. Dodatkowo, funkcja `zwizualizuj_emocje` wyświetla fale dźwiękowe dla różnych emocji na podstawie wybranych plików dźwiękowych.

Przykład użycia dla nowego pliku

Na zakończenie implementacji, przetestowano metode na nowym pliku dźwiękowym. Wczytano dane, przewidziano emocje, a następnie zwizualizowano fale dźwiękowe dla tego pliku.

W kolejnych etapach projektu planujemy dalsze dostrojenie modelu oraz ewentualne wprowadzenie kolejnych metod w celu poprawy skuteczności rozpoznawania emocji.