Przetwarzanie i Rozpoznawanie Dźwięku

Rozpoznawanie stylów tanecznych na podstawie nagrań muzycznych

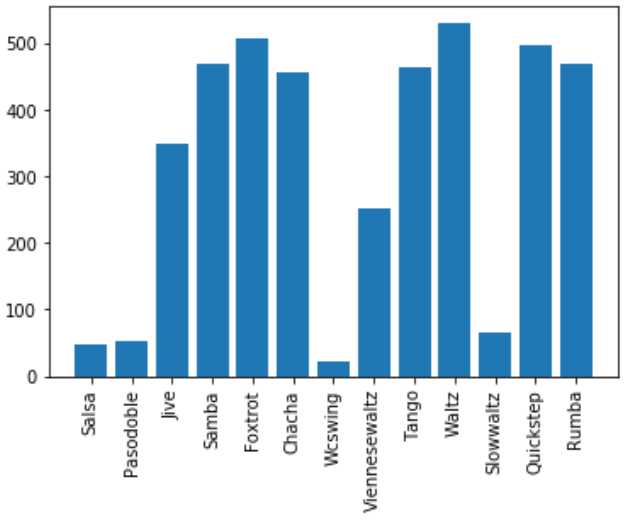
1. Wstęp

Celem projektu było rozpoznawanie stylów tanecznych na podstawie ich nagrań muzycznych.
Do implementacji wykorzystano język ***Python3*** z bibliotekami:

- ***Librosa*** - funkcja MFCC
- ***SoundFile*** - wczytywanie plików jako sygnał i częstotliwość
- ***Sklearn*** - klasyfikator KNN
- ***TensorFlow*** - framework obsługujący sieci neuronowe
- ***Keras*** - API usprawniające pracę z TensorFlow

2. Dane

Dane pochodziły ze zbioru Extended Ballroom. Zawiera on 4000 30 sekundowych piosenek, podzielonych na 13 stylów: salsa, pasodoble, jive, samba, foxtrot, cha-cha, swing, walc wiedeński, tango, walc angielski, quickstep oraz rumba. Dane są silnie niezbalansowane i niektóre style posiadają zaledwie parędziesiąt piosenek. Rysunek 1 przedstawia histogram liczebności klas.

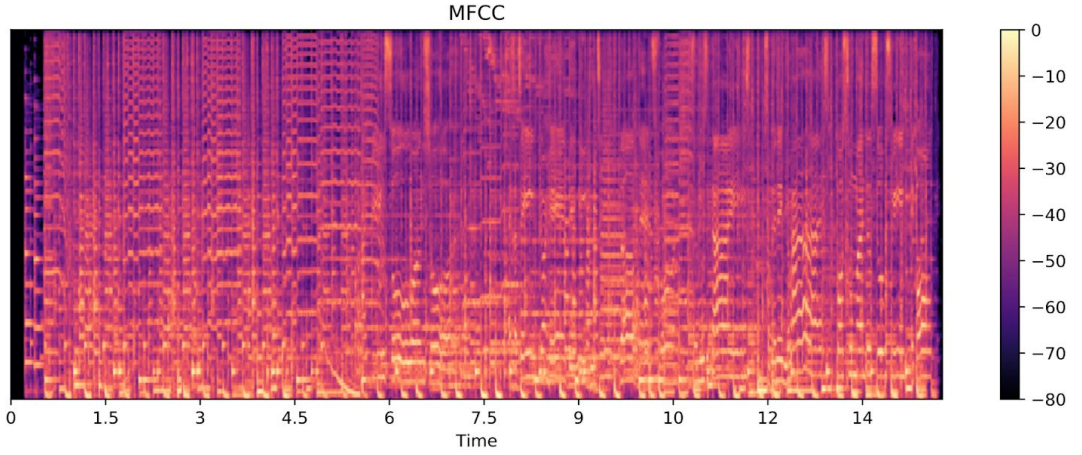


Rys. 1 Rozkład klas w zbiorze danych

3. Preprocessing

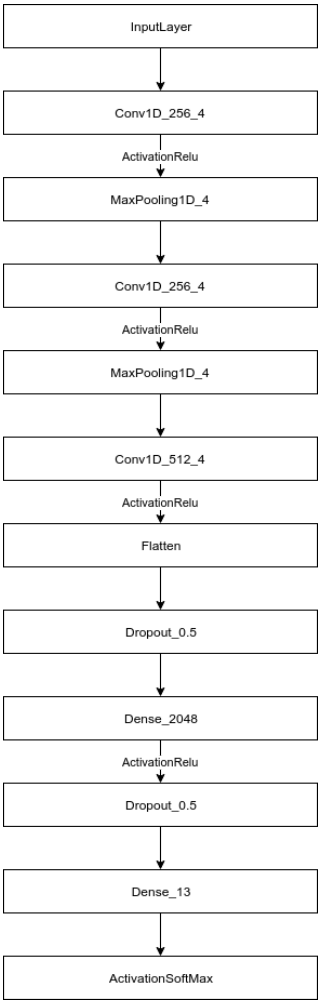
Do wstępnego przetworzenia danych posłużyła nam biblioteka **librosa**. Wszystkie piosenka zostały wczytane i przeprowadzono na nich analizę MFCC z parametrami n_fft=2048, n_mels=128, hop_length=1040. Następnie wyznaczono spectrogram obcięty do długości 636, który ponownie zapisano w formacie .npy, aby umożliwić ponowne szybkie wczytanie za

pomocą biblioteki numpy. Tak przetworzone dane posłużyły jako dane wejściowe. Ponieważ preprocessing trwał dość długo, została użyta biblioteka ***joblib***, aby zrównoleglić przetwarzanie danych.



Rys 2. Przykładowy spektrogram

Rys 3. Użyta architektura

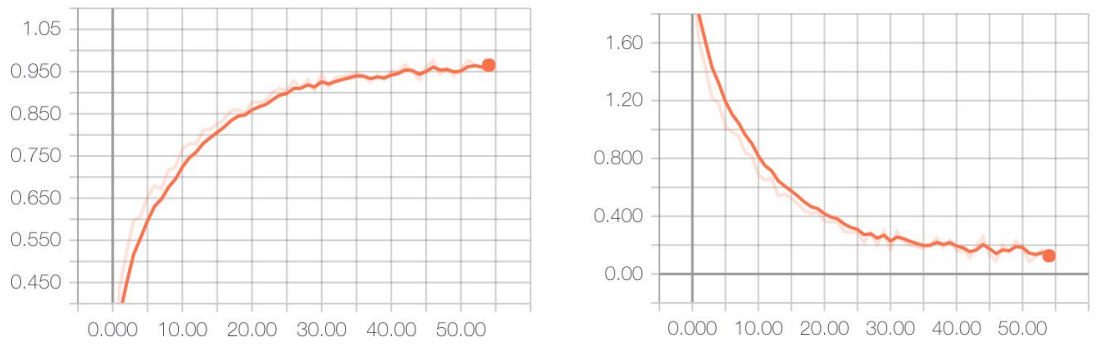


4. Metoda klasyfikacji

Do wydobycia cech oraz klasyfikacji została użyta konwolucyjna sieć neuronowa która enkodowała na wyjściu style za pomocą hot-end encoding. Na jej wejście została wprowadzana tablica o wymiarach 636 x 128. Następnie zostały użyte trzy jednowymiarowe warstwy konwolucyjne z odpowiednio 256, 256 i 512 filtrami w przestrzeni czasu o szerokości 4 z odstępem 2. Następnie wykorzystano dwie warstwy gęste z regulacją dropout. Wszystkie warstwy używały aktywacji ReLu. Na końcu architektury użyto warstwy gęstej z 13 wyjściami i aktywacją softmax.

5. Proces uczenia

Przeprowadziliśmy uczenie batchowe po wczytaniu wszystkich danych do pamięci RAM. Po obserwacji krzywej uczenia dobraliśmy ilość epok równą 20 oraz rozmiar batchu wynoszący 32. Uczenie trwało ok 4 minuty, czyli 12 sekund na epokę, co pozwoliło osiągnąć loss wynoszący 0.0951 oraz accuracy równe 0.9718 na zbiorze testowym.



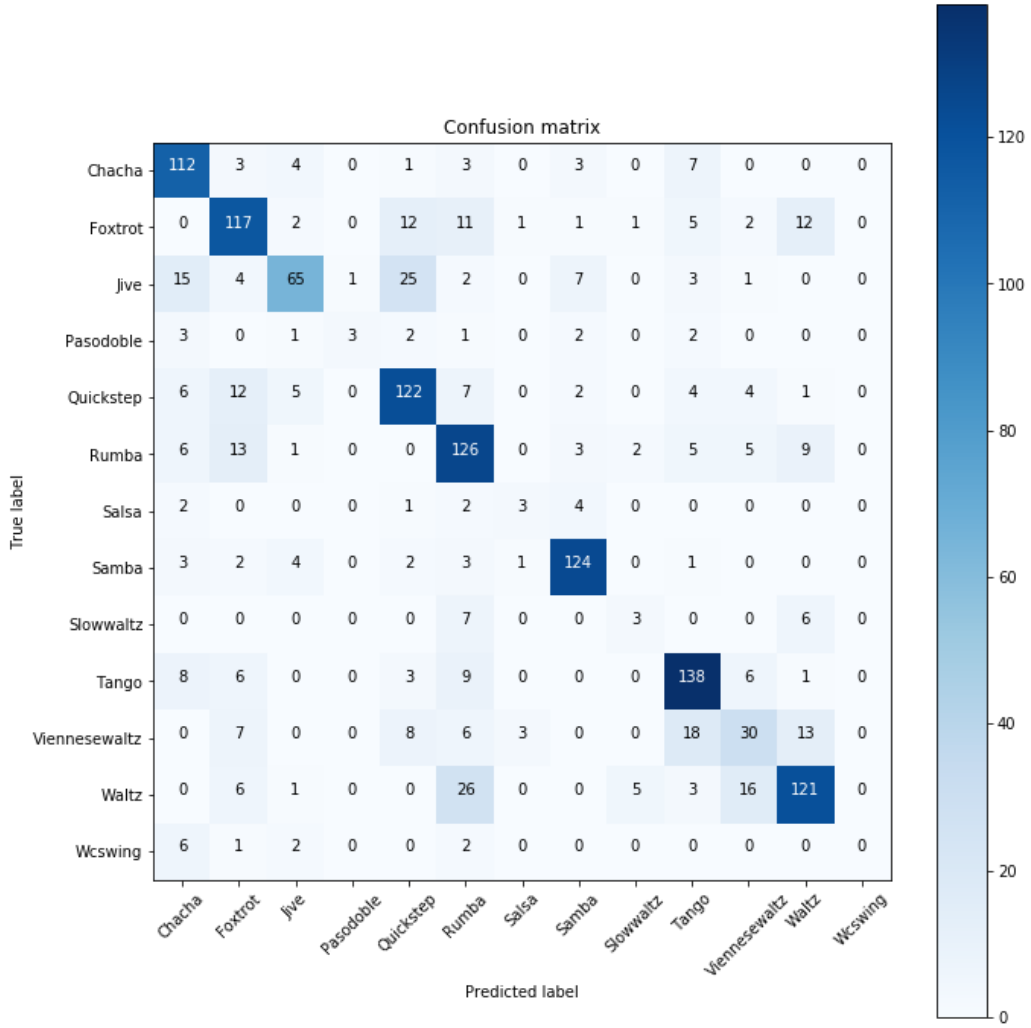
Rys 4. Wykresy accuracy (po lewej), loss (po prawej)

6. Macierz pomyłek

Dane zostały podzielone na dwa rozłączne podzbiory:

- treningowy - ⅓ danych
- testowy - ⅔ danych

Rysunek 5 przedstawia działanie klasyfikatora.



Rys 5. Macierz pomyłek klasyfikatora

Klasyfikator rozpoznał rodzaje muzyki tanecznej z 69,85% dokładności. Z macierzy pomyłek widać że najczęstsze błędy popełniał pomiędzy stylami Quickstep oraz Jive, Są to bardzo podobne style których tempo wynosi ok 200 uderzeń na minutę oraz wykorzystują tego samego rodzaju instrumenty. Podobnie problemy pojawiły się z klasami które posiadały mało przykładów.

7. Porównanie z innymi pracami

Poszukiwaliśmy także innych artykułów które badały wybrany dataset. I tak na przykład w artykule “Randomly weighted CNNs for (music) audio classification”, Jordi Pons oraz Xavier Serra użyli MFCC oraz SVM by stworzyć podstawowy model do klasyfikacji rytmu i tempa. W pracy badali oni czy sieci konwolucyjne mogę polepszyć uzyskaną przez SVM klasyfikację. Jest to nieco inne zagadnienie gdyż styl piosenki nie jest zawsze ściśle związany z tempem nagrania. Poza tym zadanie jakie sobie postawili jest dużo łatwiejsze. Przedstawiony w pracy podstawowy model osiągnął 65.49% skuteczności, a użycie sieci konwolucyjnych pozwoliło poprawić ten wynik. Jednak nie obeszło się bez użycia ogromnych sieci jak VGG, aby to osiągnąć.

8. Wnioski

Sieci konwolucyjne dobrze sprawdziły się przy klasyfikacji stylu tanecznego muzyki.

9. Bibliografia

- Deep Music Genre, Miguel Flores Ruiz de Eguino http://cs231n.stanford.edu/reports/2017/pdfs/22.pdf?fbclid=IwAR1PfgYrw2ZYB4P4KwlsQ0RSQBrB3lwJZTCSwjSOiTxk9ByhqEUIeaGvq3M
- Music Genre Classification, Matan Lachmish https://medium.com/@matanlachmish/music-genre-classification-470aaac9833d?fbclid=IwAR2jCm6t6n65psstkYba8lr_GoygCPcST0UQ8mJYnE19vMpwejvc89t8kqs
- Randomly weighted CNNs for (music) audio classification, Jordi Pons, Xavier Serra https://arxiv.org/abs/1805.00237