Rozpoznawanie wypowiedzi z wykorzystaniem DTW

PIRD 2018

Prowadzący: dr inż. Ewa Łukasik

1. Wstęp

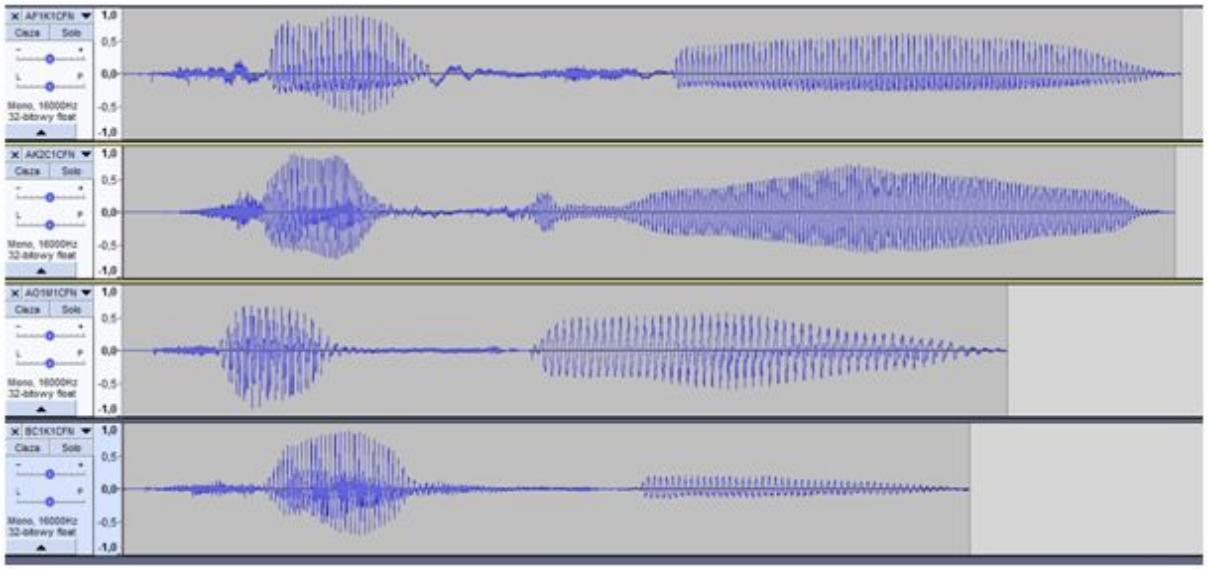
Celem zadania było analiza i stworzenie programu umożliwiającego porównywanie wypowiedzianych przez różne osoby komend oraz sprawdzenie, z jaką trafnością da się klasyfikować te komendy. W rozpoznawaniu wykorzystano metodę MFCC, która zwraca współczynniki kepstralne dla danego sygnału oraz procedurę DTW, dzięki której możliwe było porównanie dwóch sygnałów w czasie.

W projekcie skorzystano z bibliotek:

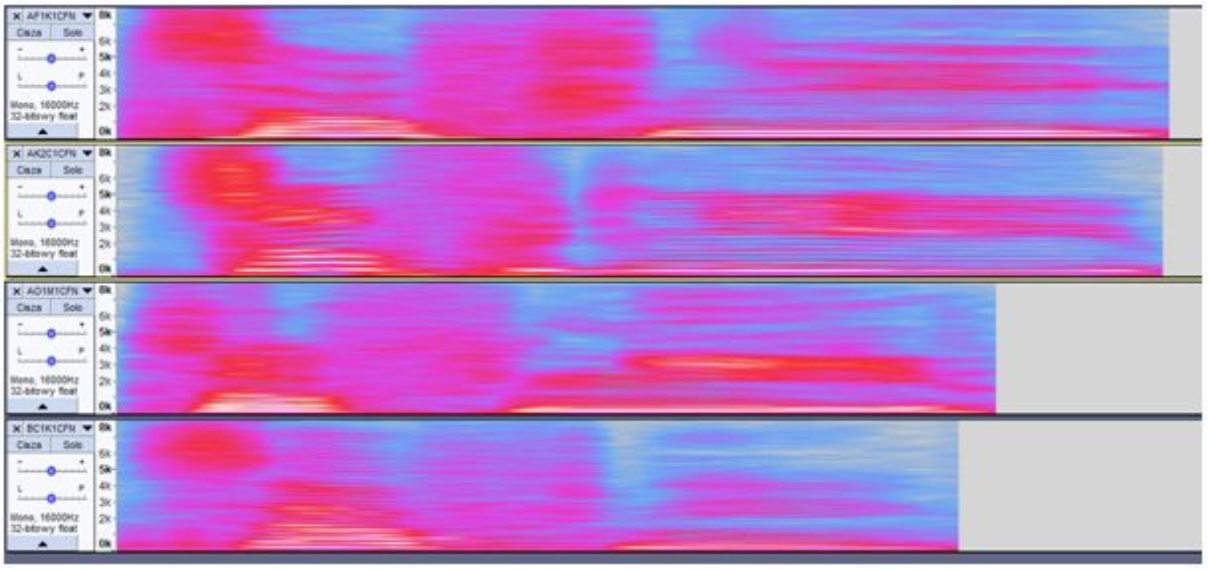
- Librosa, która dostarcza funkcję MFCC,
- DTW, która dostarcza funkcję DTW,
- SoundFile, która wczytuje pliki jako sygnał oraz częstotliwość próbkowania

Program zaimplementowano w języku Python.

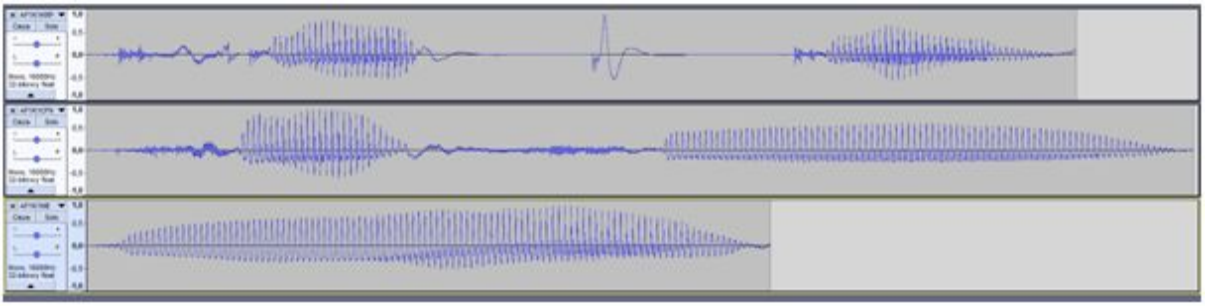
2. Obserwacja przebiegów czasowych, widm i spektrogramów wypowiedzi



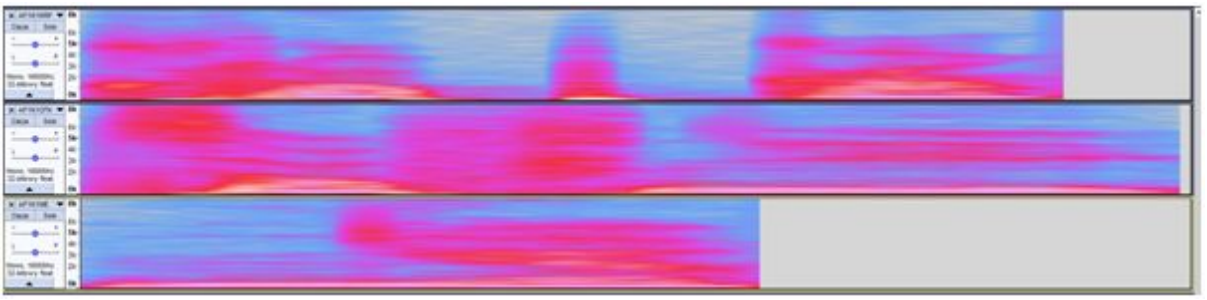
Rys 1. Przebieg czasowy dla wypowiedzi: cofnij - wypowiedziane przez cztery różne osoby



Rys 2. Spektrogram dla wypowiedzi: cofnij - wypowiedziane przez cztery różne osoby



Rys 3. Przebieg czasowy dla wypowiedzi: cofnij, kropka, nie -
wypowiedziane przez jedną osobę



Rys 4. Spektogram dla wypowiedzi: cofnij, kropka, nie - wypowiedziane przez jedną osobę

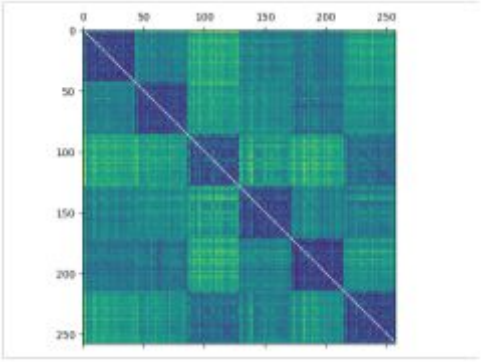
Przed przystąpieniem do implementacji, przeanalizowano dostępne komendy – ich przebiegi czasowe oraz spektrogramy. Można zauważyć, że ta sama wypowiedź wypowiedziana przez różne osoby jest bardzo podobna w przebiegu czasowym (Rys. 1) , choć nie identyczna. Różnice wynikają ze sposobu mówienia mówcy, tonu i rodzaju głosu oraz ewentualnego wzmocnienia pewnej części wypowiedzi. Spektrogram pokazuje jednak, że w większości przypadków ta sama wypowiedź wypowiedziana przez różne osoby ma jednak podobną charakterystykę (Rys. 2).

Widoczne jest charakterystyczne krótkie wzmocnienie do częstotliwości 2kHz w pierwszej fazie oraz dłuższe, do częstotliwości 1 kHz w drugiej.

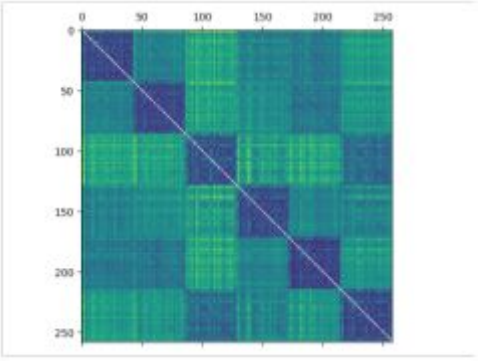
Problemem mogą być jednak różne wypowiedzi wypowiedziane przez tę samą osobę (Rys. 3). Pokazuje on, że niektóre wypowiedzi choć są różne to ich przebiegi czasowe mogą być dosyć podobne (przebieg nr 1 i nr 2 na Rys. 3). Spektogram (Rys. 4) sygnału nr 1 i nr 2 ma podobne charakterystyczne wzmocnienia, choć wypowiedziane komendy nie są takie same. Sygnał nr 3 odróżnia się jednak znacznie od sygnału nr 1 i nr 2 w przebiegu czasowym jak i na wykresie spektrum.

3. Odpowiednia długość ramki i liczba współczynników MFCC

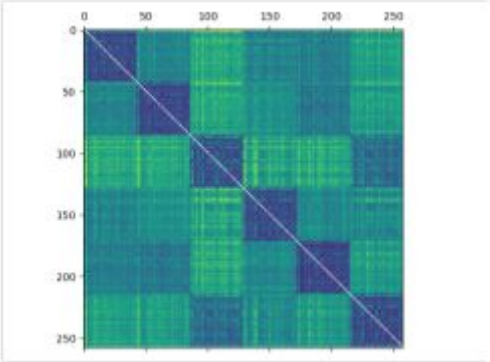
Długość ramki dobrano w sposób eksperymentalny, na podstawie mapy podobieństwa odległości. Widać, że wraz ze wzrostem długości ramki, mapa przyciemnia się, a więc wartości poza główną przekątną stają się bliższe, co jest skutkiem niepożądanym. Najlepsze efekty uzyskano dla długości ramki równej 256, dla 128 jakość klasyfikacji może się znowu pogarszać. Poniżej przedstawiono mapy podobieństwa dla różnych długości ramki.



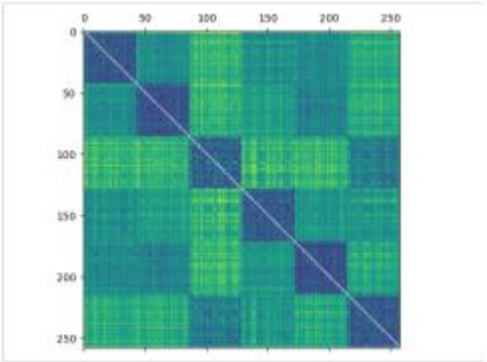
Rys 5. 2048



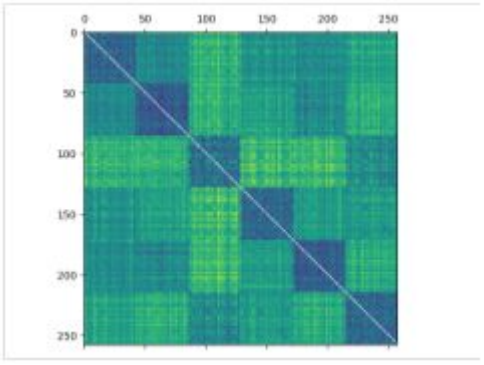
Rys 6. 1024



Rys 7. 512



Rys 8. 256



Rys 5. 128

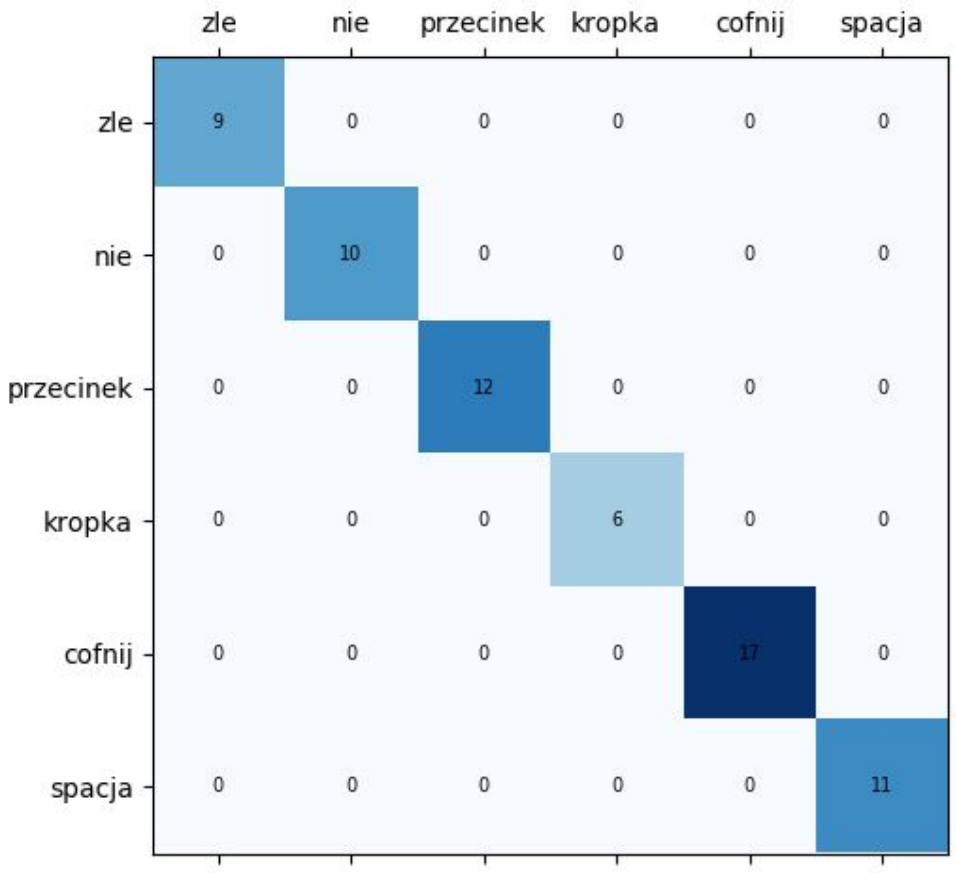
Dodatkowo, liczba współczynników kepstralnych, która następnie była porównywana w metodzie DTW została ustalona na 20.

4. Procedura i wyniki rozpoznawania

Wraz ze zmianą parametrów, udało się uzyskać trafność klasyfikacji na poziomie 100%. Zbiór wszystkich komend został podzielony na nagrania treningowe oraz testowe w proporcjach 3:1. Następnie każda poddawana testowaniu wypowiedź, została porównana z próbkami ze zbioru treningowego i wynikiem klasyfikacji była próbka o najmniejszej odległości od testowanej próbki.

Funkcja kosztu to suma odchyłek od prostej liniowej reprezentującej ścieżkę – dla dwóch identycznych nagrań, ścieżka jest funkcją liniową a odległość wynosi 0. Należy jednak zauważyć, że nie zawsze każde nagranie z tej samej grupy komend, było bliżej niż rozwiązanie z innej grupy tzn. czasami komenda „nie” wypowiedziana przez pewną osobę była bardziej podobna do komendy „źle”, niż inne wypowiedzi komendy „źle” przez pozostałe osoby.

Rys. 6. przedstawia macierz błędów czyli przewidziane przez napisany algorytm klasy dla zbioru testowego. Każda z komend została prawidłowo rozpoznana i przydzielona do odpowiedniej klasy.



Rys 6. Macierz błędu dla rozpoznawanych komend

5. Wnioski

Przetwarzanie mowy ludzkiej jest ważnym zagadnieniem dzięki któremu możemy komunikować się z komputerami. Wykorzystując tak zaawansowane procedury jak MFCC i DTW osiągnęliśmy 100% trafności w rozpoznawaniu mowy. Kolejne badania nad analizą sygnałów na pewno przyniosą zwiększeniem inteligencji otaczających naszych urządzeń oraz łatwiejszej komunikacji z nimi.