

Vysoké učení technické v Brně
Fakulta informačních technologií

Signály a systémy

Query by example

Martin Kostelník (xkoste12)

xkoste12@stud.fit.vutbr.cz

16.12.2019

Úkol 1

Tabulka obsahuje názvy nahrávek a jejich délku ve vzorcích a v sekundách.

Název	Délka [n]	Délka [s]
sa1	63200	3.95
sa2	57440	3.59
si1055	92960	5.81
si1685	65360	4.08
si2315	45680	2.85
sx65	51200	3.20
sx155	56000	3.50
sx245	43520	2.72
sx335	87200	5.45
sx425	39440	2.46

Nahrávky můžete použít dle libosti. V zadání varianta C.

Úkol 2

Z nahraných vět jsem vybral následující slova:

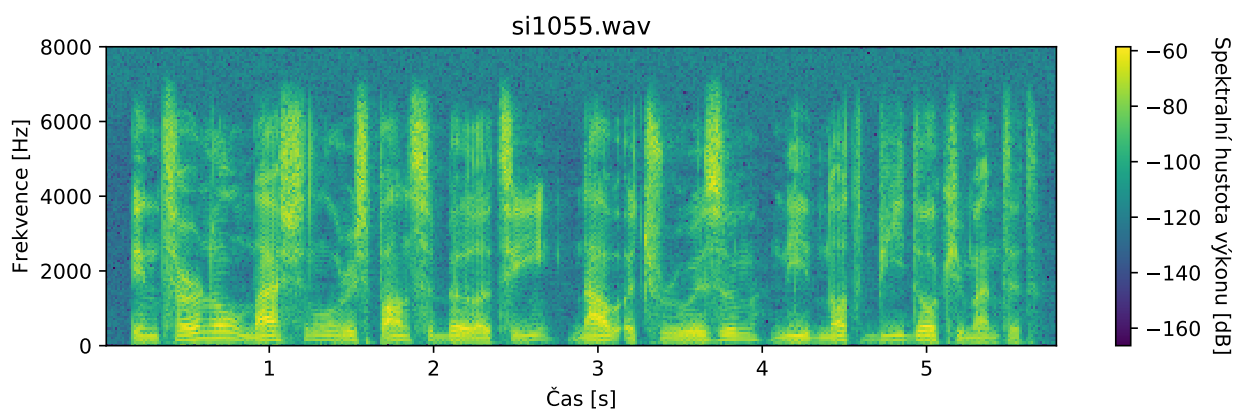
- responsibility (q1)
- intelligence (q2)

V tabulce je uveden název souboru, jeho délka ve vzorcích a v sekundách.

Název	Délka [n]	Délka [s]
q1	18800	1.18
q2	12560	0.79

Úkol 3

Spektrogram věty si1055:



Pro generování spektrogramu je v kódu implementována funkce `generateSpectrogram(fileName)`. Funkce bere jako parametr cestu k nahrávce, ze které se má spektrogram vygenerovat. Po zavolání funkce se v aktuálním adresáři vytvoří soubor `spectrogram.pdf` s vygenerovaným spektrogramem.

Úkol 4

K výpočtu parametrů (features) jsem implementoval funkci `getFeatures(fileName)`. Funkce má jeden parametr, opět je to cesta k nahrávce. Funkce pro každý rámec vyprodukuje 16 koeficientů z logaritického výkonového spektra, což je matice o 256 řádcích. Koeficient se spočítá tak, že se sečte vždy 16 po sobě jdoucích řádků v daném rámci. Výsledek této funkce je tedy matice $16 \times N$ (N se liší dle délky nahrávky). Implementovaná funkce vrací navíc ještě čas nahrávky, který je potřeba později při vykreslování grafů.

Úkol 5

Výpočet klíčového skóre slova pro jednu větu zprostředkovává funkce *getRating(sentence, query)*. Tato funkce má dva parametry. První je cesta k nahrávce věty a druhý je cesta k nahrávce slova. Funkce vrací pole s hodnotami skóre.

Pro ilustraci přikládám kód funkce s komentáři v angličtině

```
def getRating(sentence, query):
    # Get features for both sentence and query
    # We need to transpose the matrices to match their length
    fSentence = getFeatures(sentence)[0].transpose()
    fQuery = getFeatures(query)[0].transpose()

    rating = 0
    ratingArray = []

    # Iterate the query features around sentence features and get
    # Pearson coefficients for each position of each query position
    for i in range(len(fSentence) - len(fQuery)):
        for j in range(len(fQuery)):
            rating += sp.stats.pearsonr(fQuery[j], fSentence[j + i])[0]

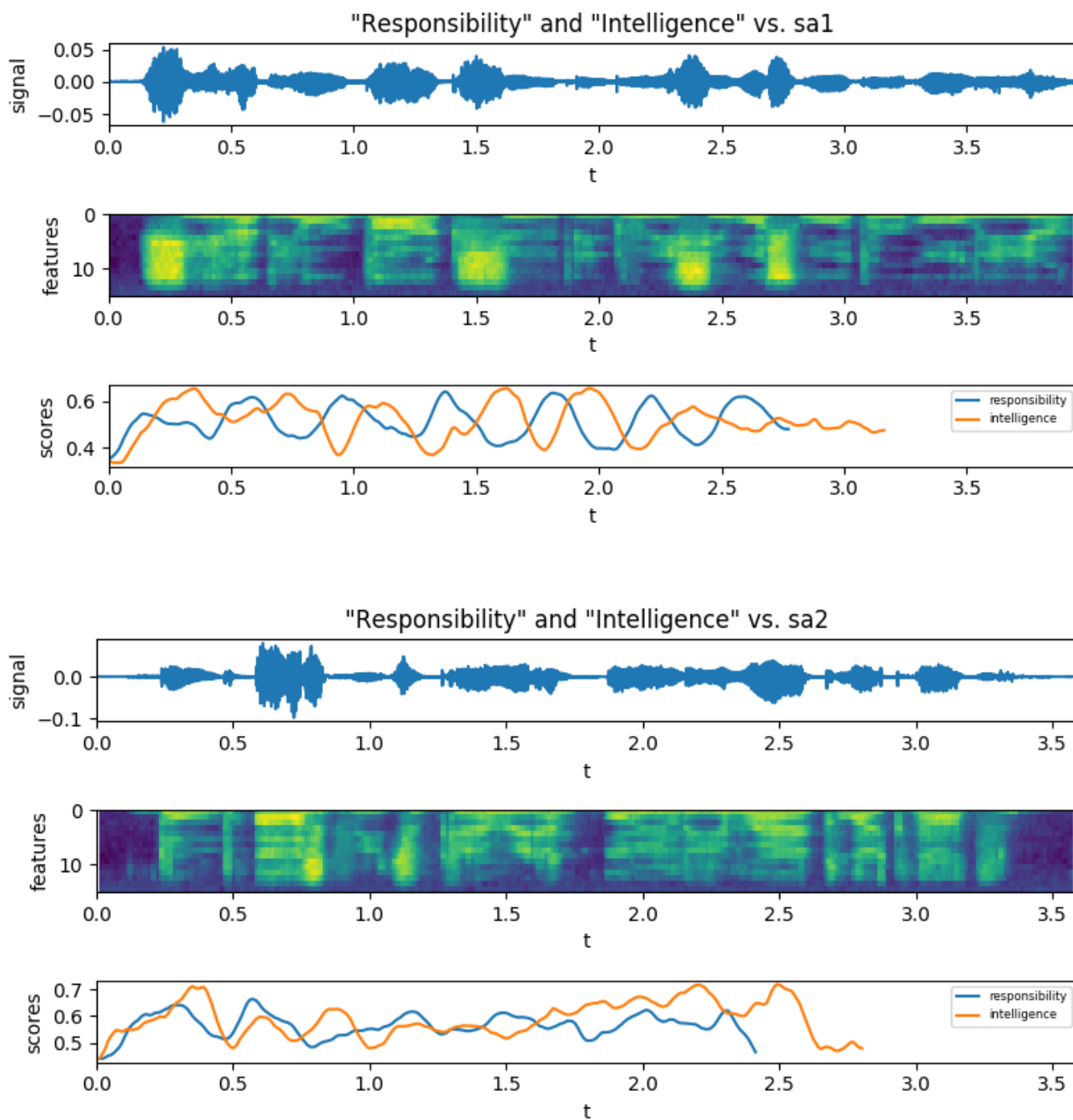
        ratingArray.append(rating)
        rating = 0

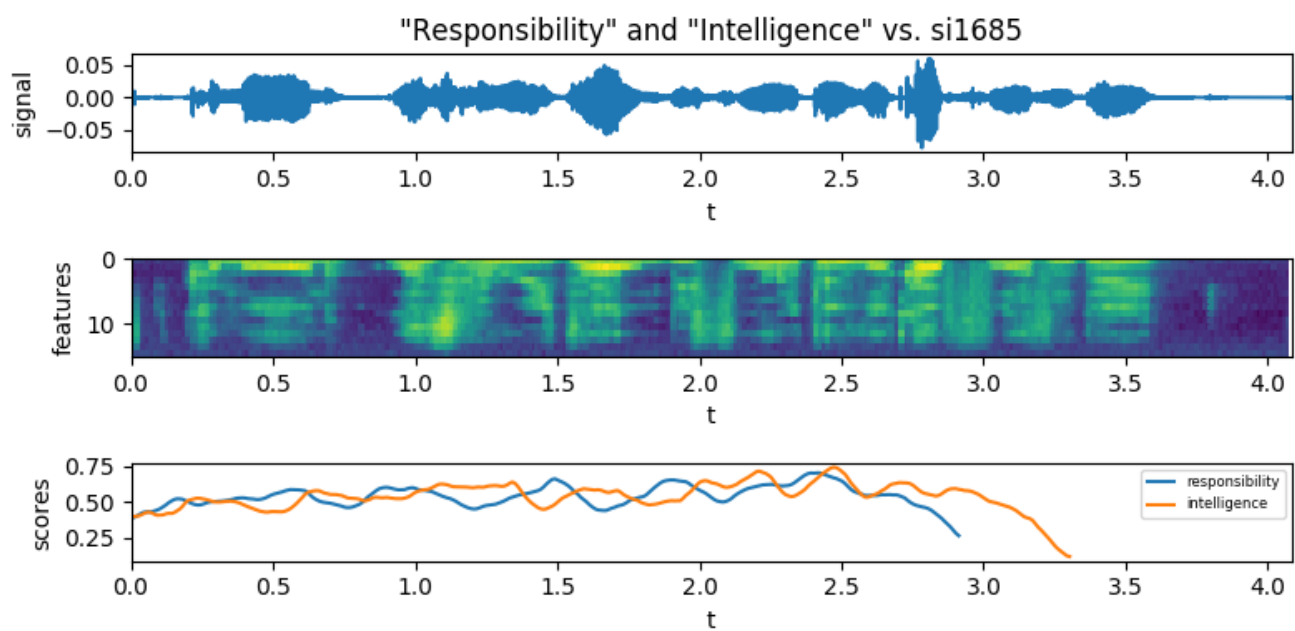
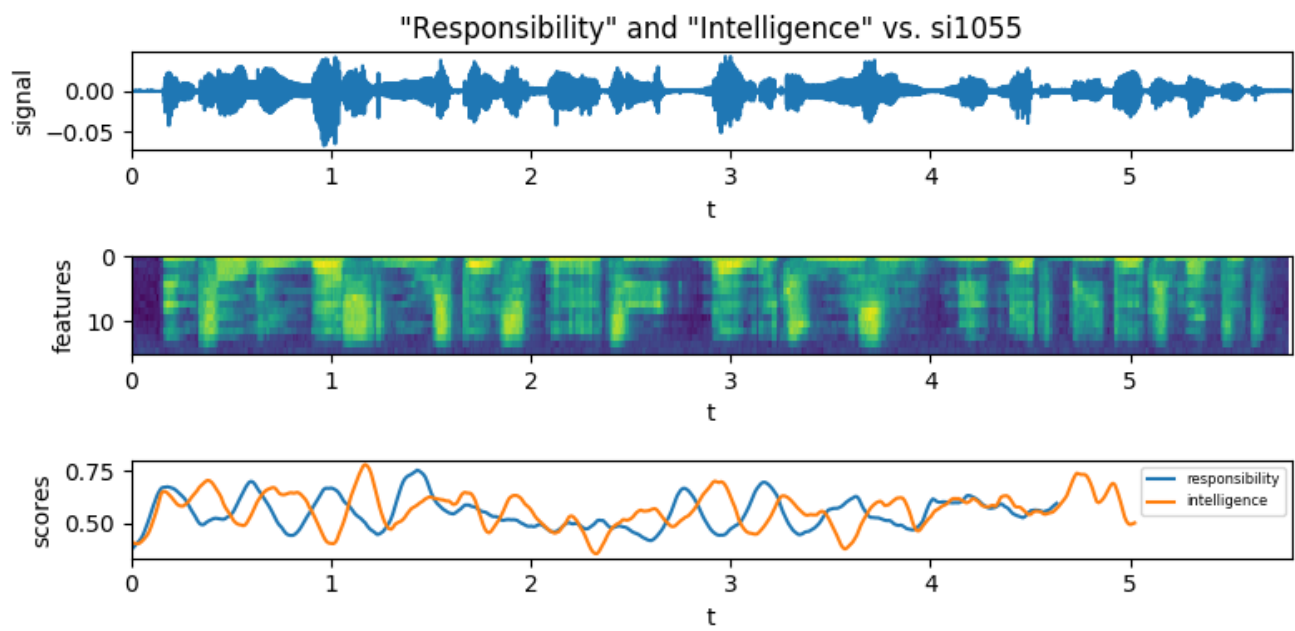
    # Divide each member of ratingArray by query length
    ratingArray[:] = [x / len(fQuery) for x in ratingArray]

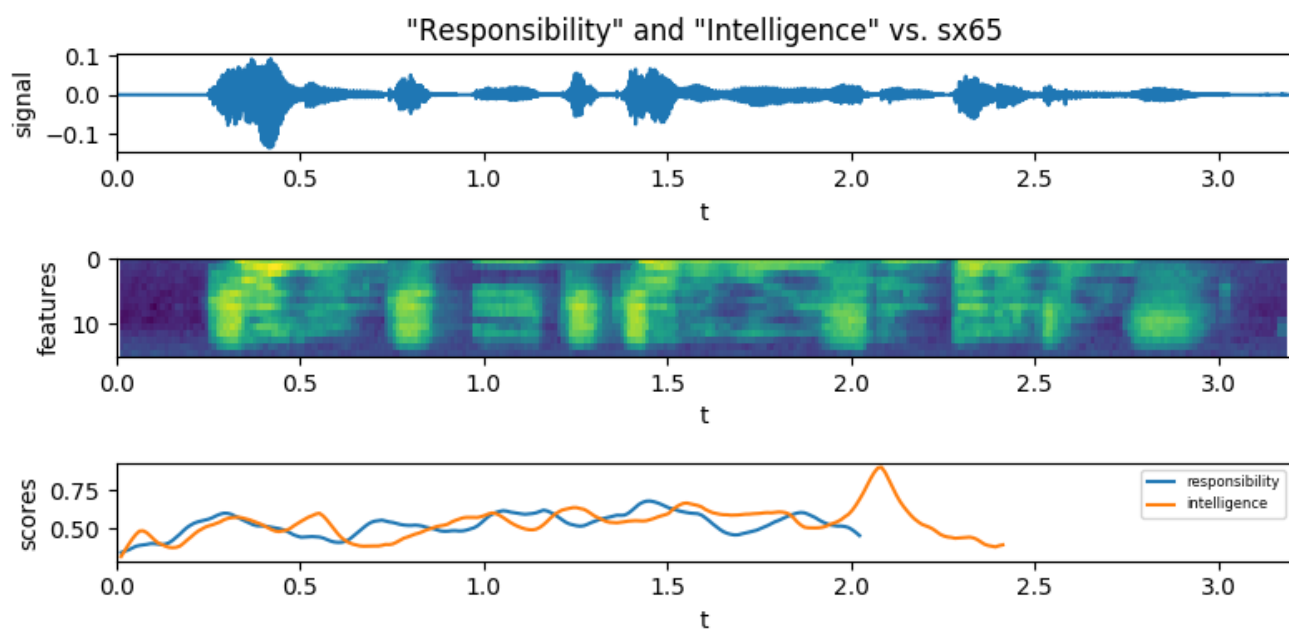
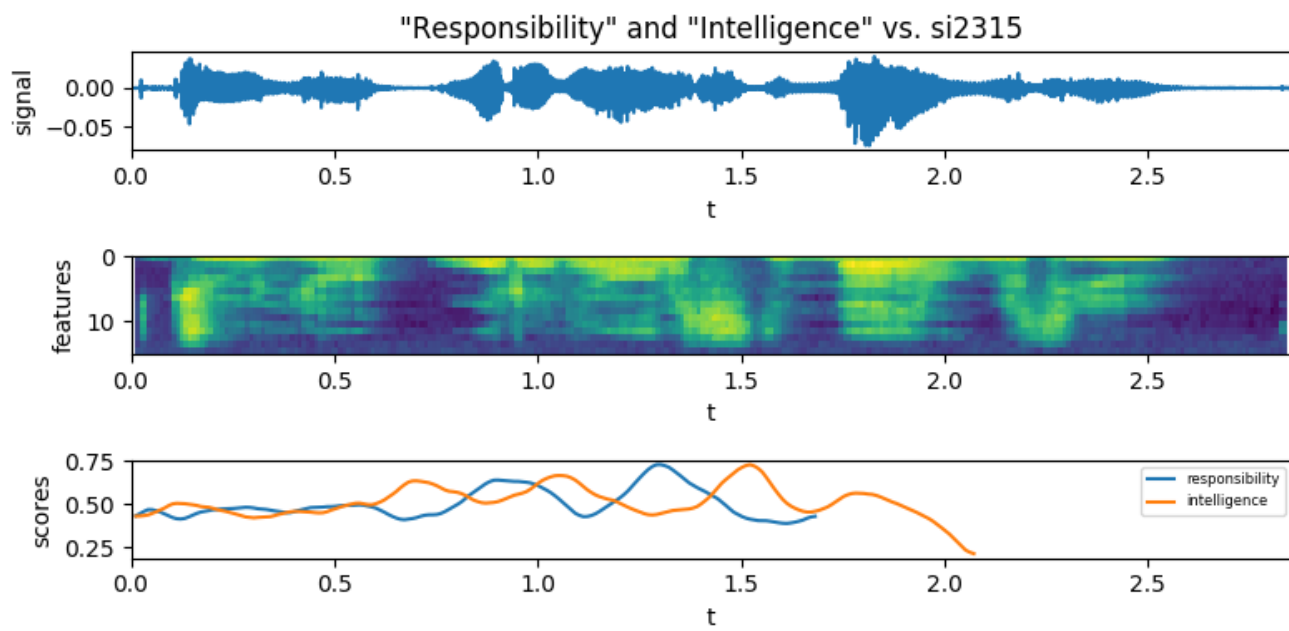
    return ratingArray
```

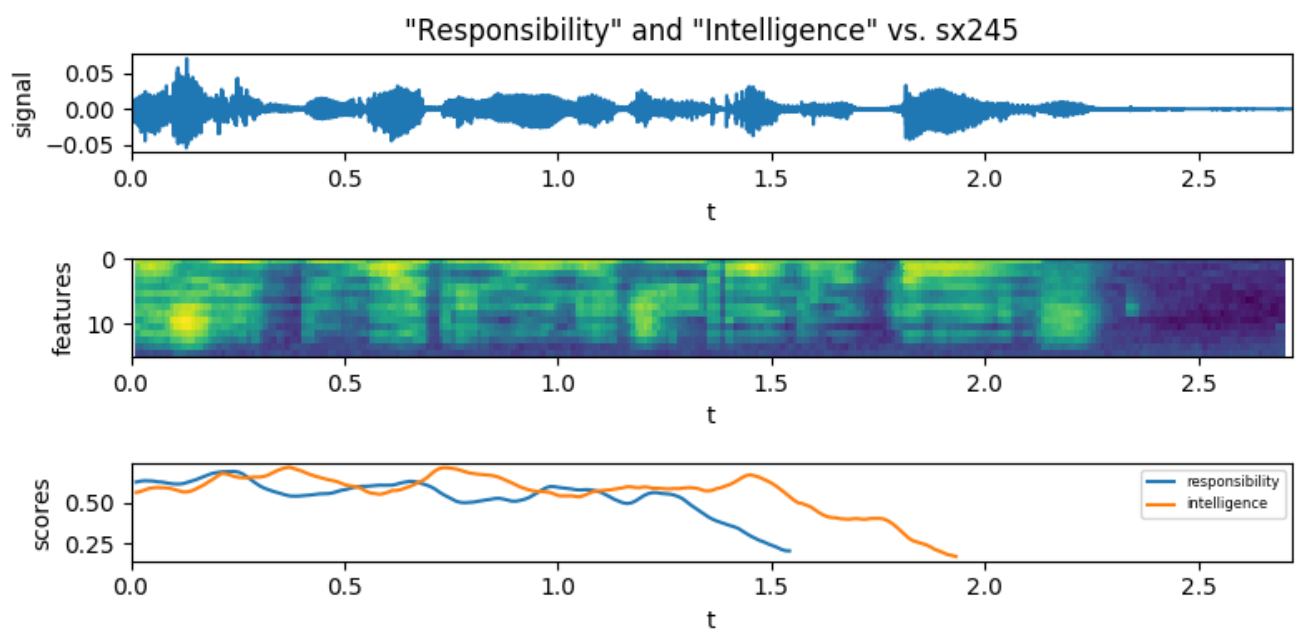
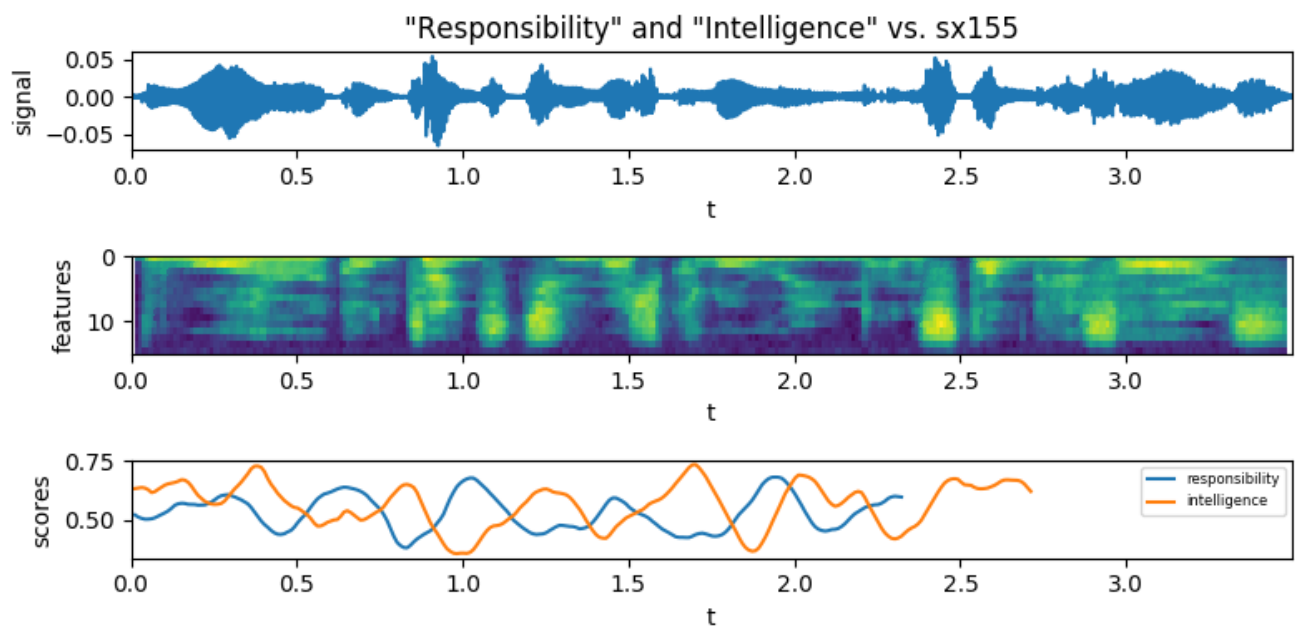
Úkol 6

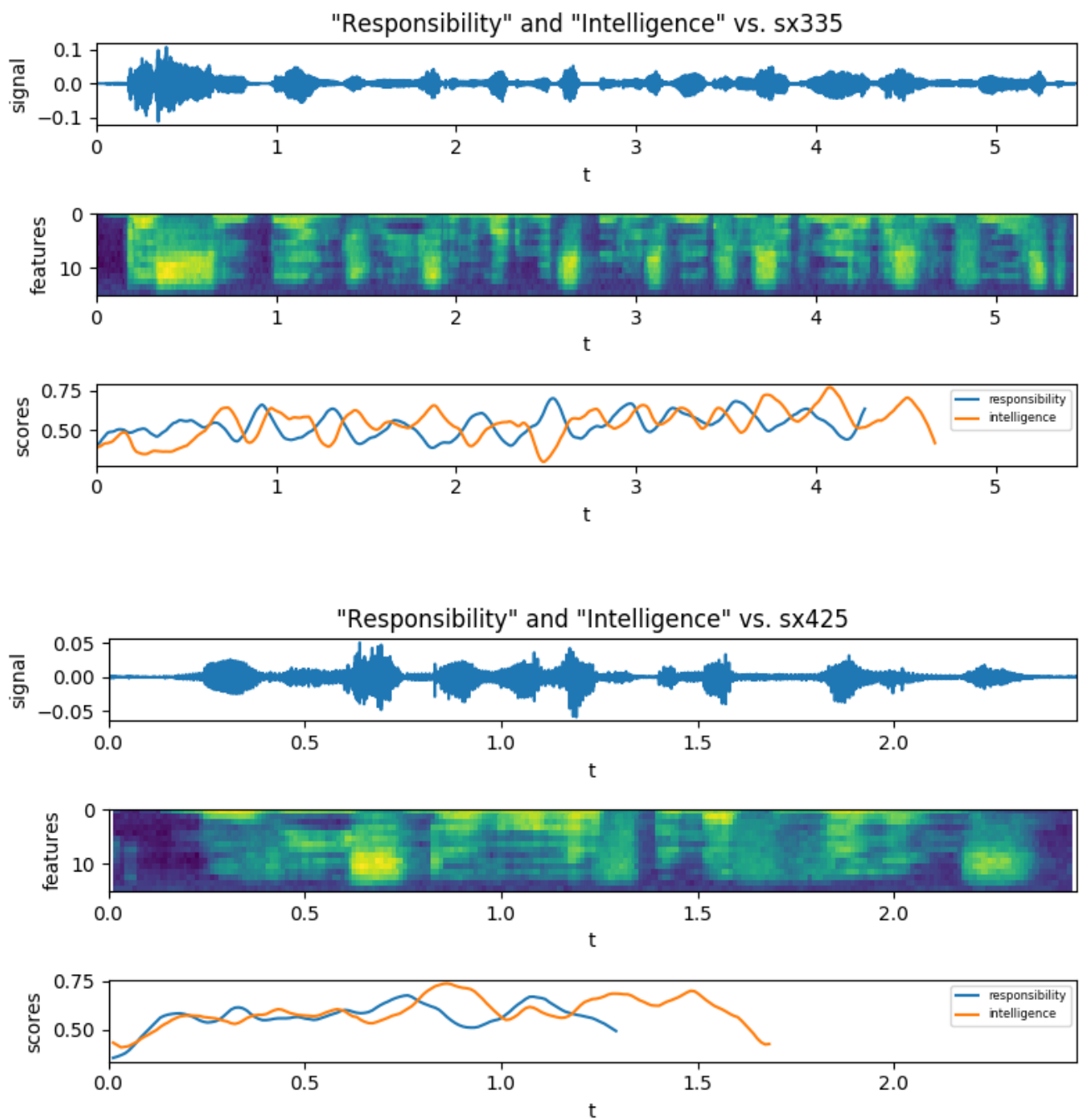
Pro generování následujících deseti obrázků slouží funkce *generateGraphs()*. Po zavolání funkce se vygeneruje deset souborů *.pdf s názvy nahrávek v aktuálním adresáři.











Úkol 7

Jelikož víme, které nahrávky obsahují vyhledávané slovo, podívali jsme se na průběhy skóre v těchto dvou nahrávkách a podle něj jsme odhadli rozhodovací práh pro každé slovo.

Práhy:

- responsibility (q1) – 0.70
- intelligence (q2) – 0.75

Úkol 8

Věta	Query	ANO / NE	Od vzorku	Do vzorku
sa1	q1	NE	-	-
	q2	NE	-	-
sa2	q1	NE	-	-
	q2	NE	-	-
si1055	q1	ANO	21960	40760
	q2	ANO	18440	37240
si1685	q1	ANO	38280	57080
	q2	NE	-	-
si2315	q1	ANO	20360	39160
	q2	NE	-	-
sx65	q1	NE	-	-
	q2	ANO	32680	51480
sx155	q1	NE	-	-
	q2	NE	-	-
sx245	q1	NE	-	-
	q2	NE	-	-
sx335	q1	ANO	40680	59480
	q2	ANO	65000	83800
sx425	q1	NE	-	-
	q2	NE	-	-

Pro generování hits jsem využil již dříve použitou funkci *generateGraphs()* kvůli zjednodušení implementace. Po zavolání této funkce je v adresáři *hits* sedm wav souborů s nálezy. Ve funkci je zakomentovaný řádek pro vytisknutí na kterém vzorku nález začíná a na kterém končí.

Úkol 9 – Závěr

Analýza všech vět i slov proběhla úspěšně, stejně jako generování spektrogramu v prvním úkolu. Funkčnost detektoru se zdá být vcelku dobrá. Bylo nalezeno celkem sedm nálezů. Dvě z nich jsou správné nálezy ve větách, kde se slova opravdu nacházejí. U jednoho z úspěšných nálezů je bohužel odsekuta poslední slabika. Dalších pět nálezů je špatných. Osobně si myslím, že je to proto, že slova v nálezech obsahují výrazné hlásky, které se objevují i v hledaných výrazech a tudíž jsou si lehce podobná.