

## Úkoly:

### Úkol 1:

Název souboru	Délka (s)	Počet vzorků
q1	0.673	10774
q2	0.940	15046
sa1	5.585	89360
sa2	5.015	80240
si753	8.375	134000
si1383	7.685	122960
si2013	5.195	83120
sx123	6.380	102080
sx213	5.375	86000
sx303	5.195	83120
sx33	4.430	70880
sx393	4.745	85920

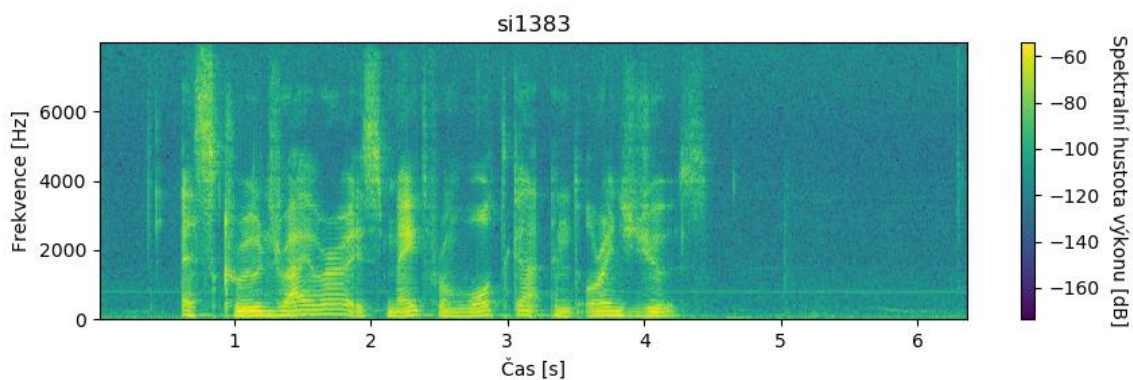
Nahrávky byly pořízeny pomocí softwaru Audacity. Klíčová slova jsou „agency“ (q1) a „screwdriver“ (q2).

Nahrávky dovoluji použít pro výzkum a vývoj v rámci řečové skupiny na FIT BUT Speech@FIT, a také pro tvorbu volně dostupné databáze „Czenglish TIMIT“.

### Úkol 2:

Querries jsou již obsaženy v tabulce v úkolu číslo 1 (první dva záznamy).

### Úkol 3:



#### Úkol 4:

Parametry jsem spočítal pomocí násobení matice A (lineární banka filtrů) s maticí se spektrogramem P. Matice A má rozměry 16x256 a je naplněna následovně:

```
78
79 # matice nul 16 x 256
80 A = np.zeros((16, 256))
81
82 # naplním matici A
83 for row in range(0, 16):
84     print_ones = False
85     printed = 0
86
87     for col in range(0, 256):
88         if col == 16 * row:
89             print_ones = True
90
91         if print_ones:
92             A[row][col] = 1
93             printed += 1
94             if printed == 16:
95                 print_ones = False
96
```

#### Úkol 5:

Výpočet skóre jsem realizoval pomocí Pearsonových korelačních koeficientů.

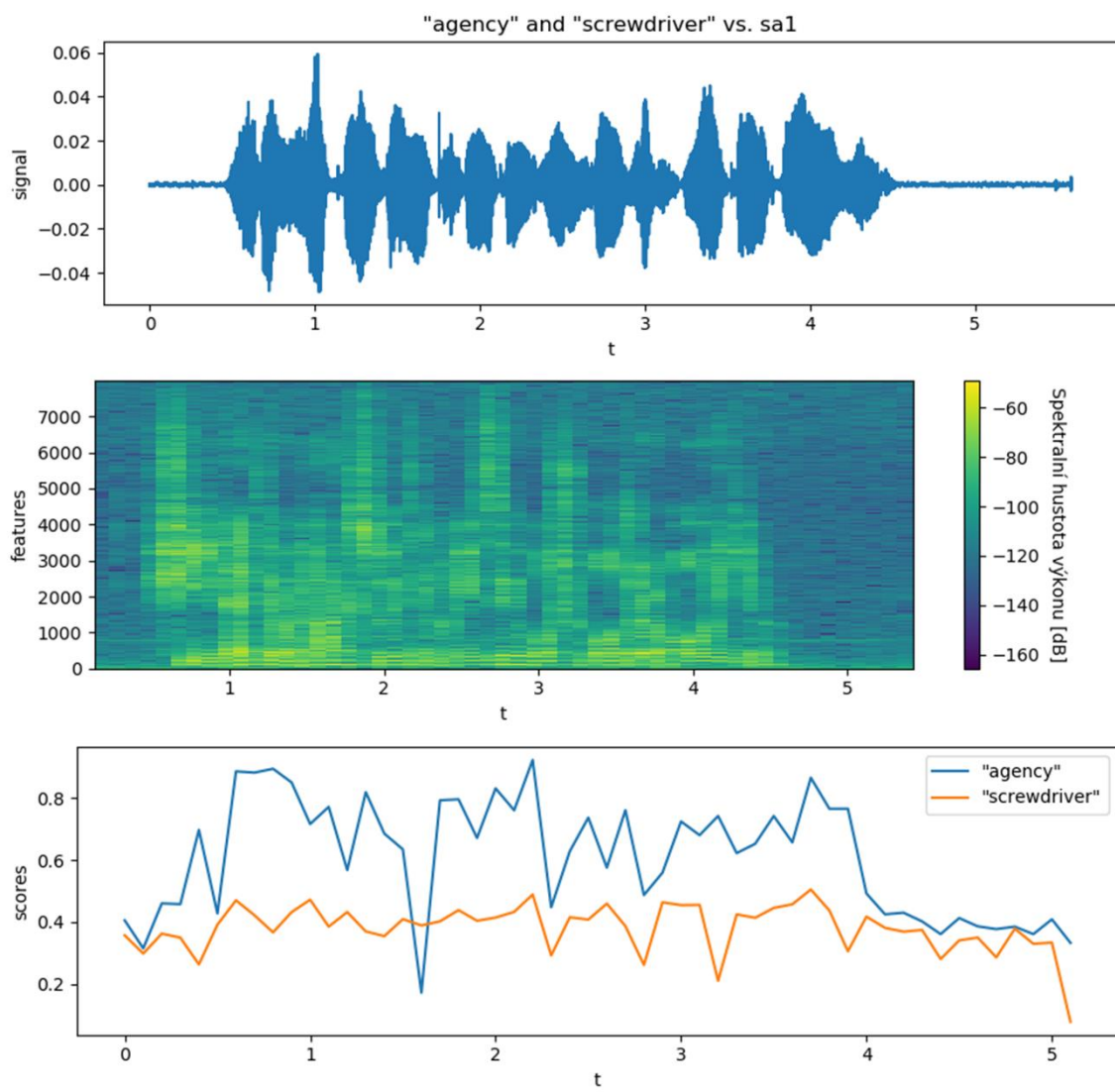
Korelační koeficienty jsou počítány pro začátek slova query, zvolil jsem prvních 10 parametrů jako referenčních, které „projíždějí“ kolem matice parametrů věty F. Referenční parametry se vždy přiloží k matici F, jsou spočítány korelační koeficienty pro úsek 10 parametrů mezi Q a F, výsledek se zprůměruje a je uložen do výsledného pole výsledků. Poté se referenční parametry přiloží k matici o jeden index dále a výpočet je opakován. Na další stránce naleznete kód pro výpočet.

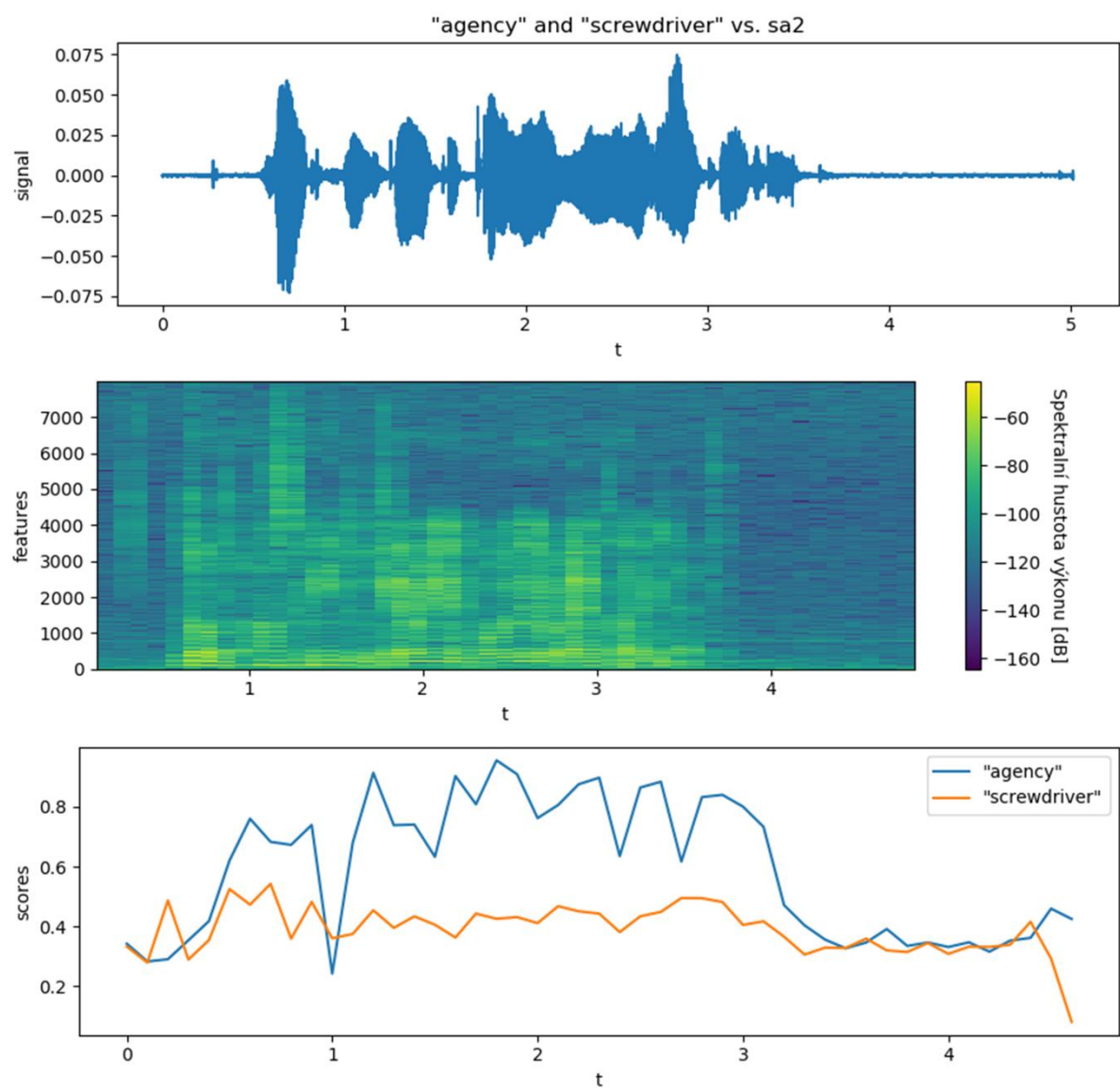
```

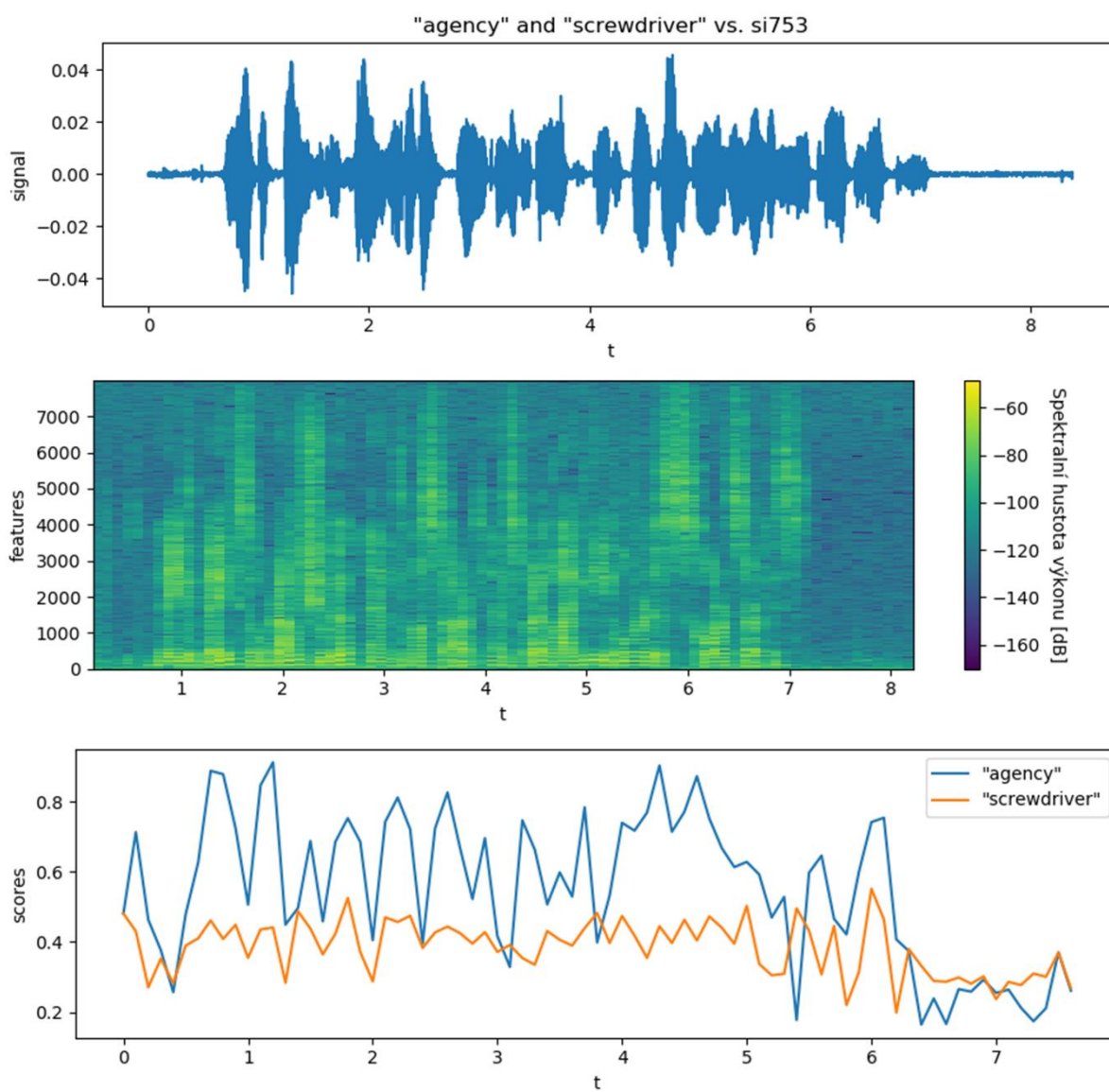
125
126     result1 = []
127     j = 0
128     index = 0
129
130     # QUERY 1
131     # VÝPOČET PRAVDĚPODOBNOSTI ZAČÁTKU QUERY VE VĚTĚ V ZÁVISLOSTI NA ČASE
132     # každý index obsahuje parametry věty, kde rozdíl mezi parametry je asi 1ms
133     while index <= t_data.size - 1:
134
135         # sem se ukládají dočasné hodnoty, které se následně průměrují
136         # (výsledek je přesnější, počítáme pravděpodobnost v každé ms)
137         temp_res = []
138
139         # j značí index s parametry v query, každá část věty se
140         # porovnává s prvními 10 parametry query (tedy začátek query)
141         while j <= 10:
142             TMP_MATRIX_F = F[:, [index]]
143             TMP_MATRIX_Q1 = Q1[:, [j]]
144             AF = np.squeeze(np.asarray(TMP_MATRIX_F))
145             AQ1 = np.squeeze(np.asarray(TMP_MATRIX_Q1))
146             r, p_value = stats.pearsonr(AQ1, AF)
147
148             # budeme brát v potaz pouze "statistically significant" hodnoty
149             r = abs(r)
150             temp_res.append(r)
151             j += 1
152             index += 1
153
154             # brání v přetečení indexu a zásahu mimo matici
155             if index + 1 >= t_data.size:
156                 break
157
158         # reset indexu query
159         j = 0
160
161         # zprůměrujeme naměřené hodnoty pro jeden úsek
162         tmp = np.mean(temp_res)
163         result1.append(tmp)
164
165

```

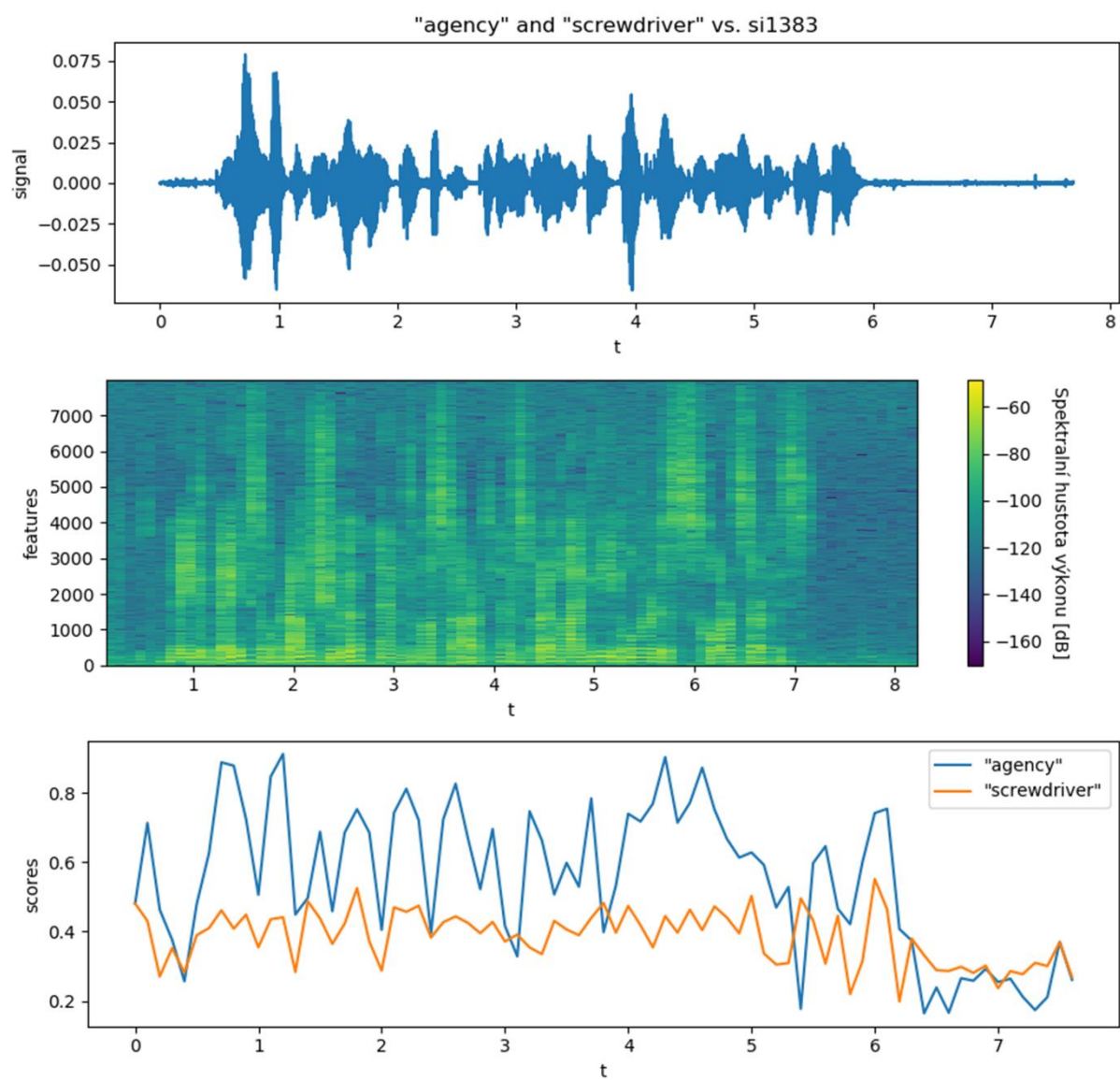
## Úkol 6:

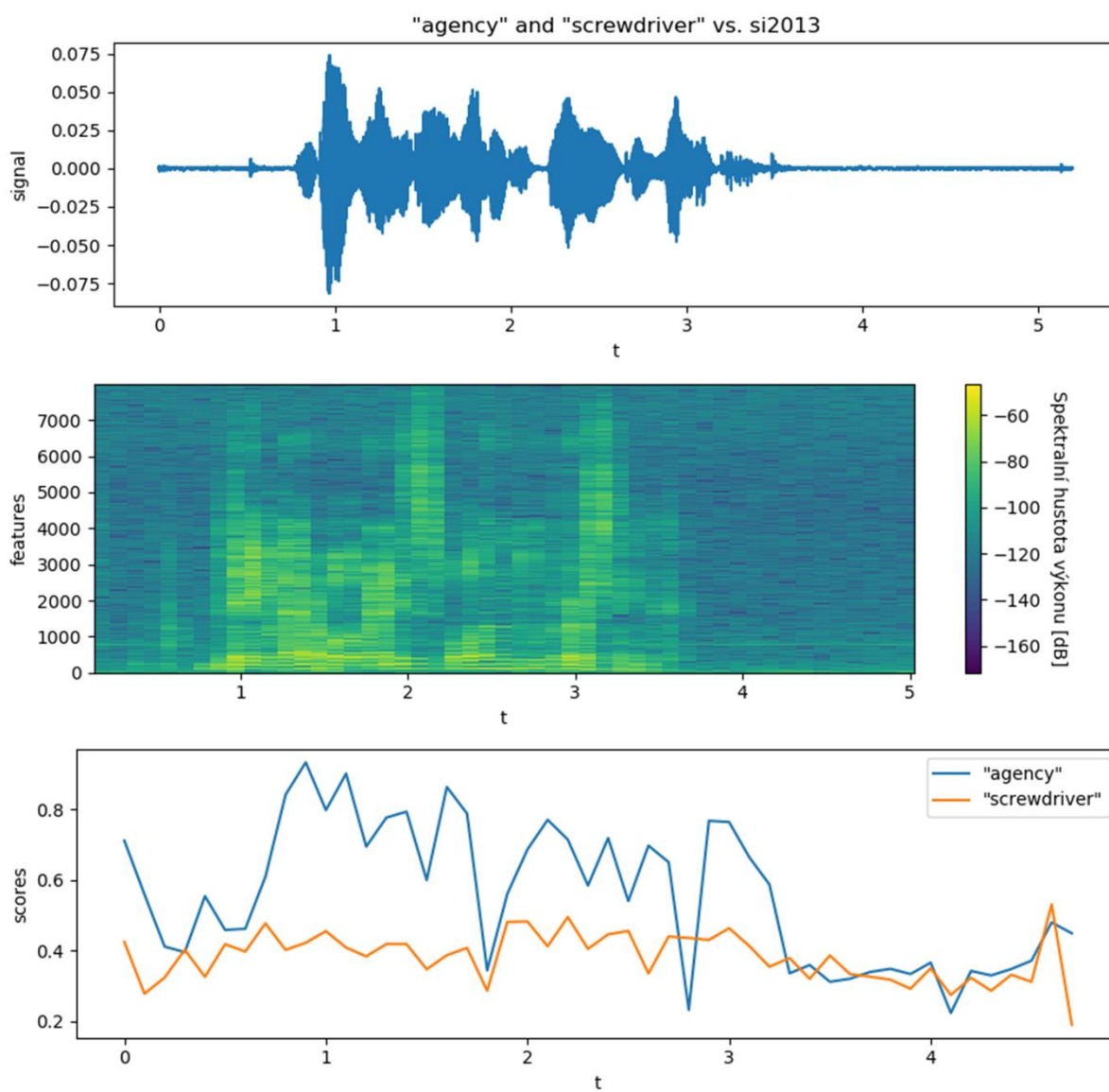




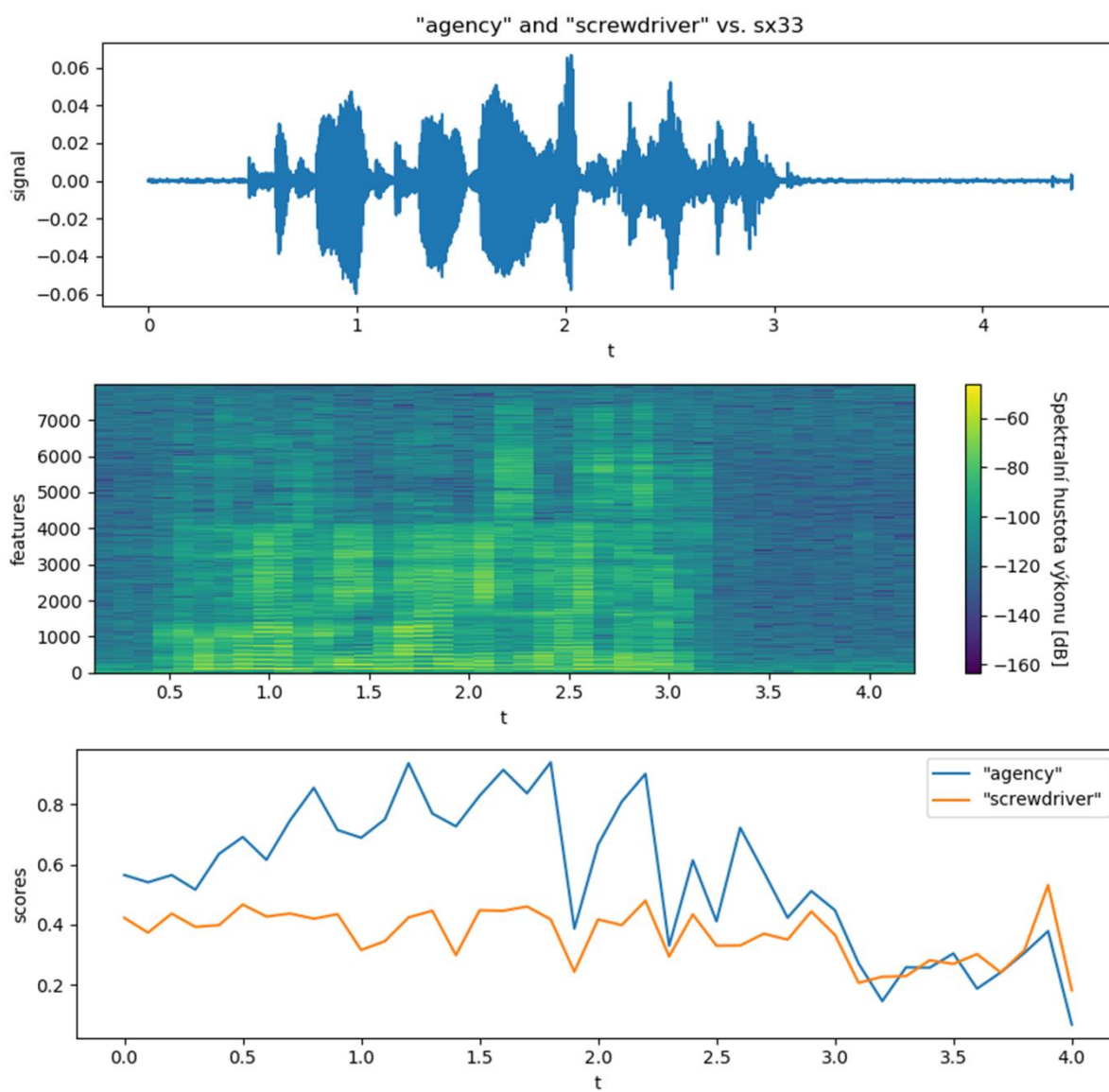


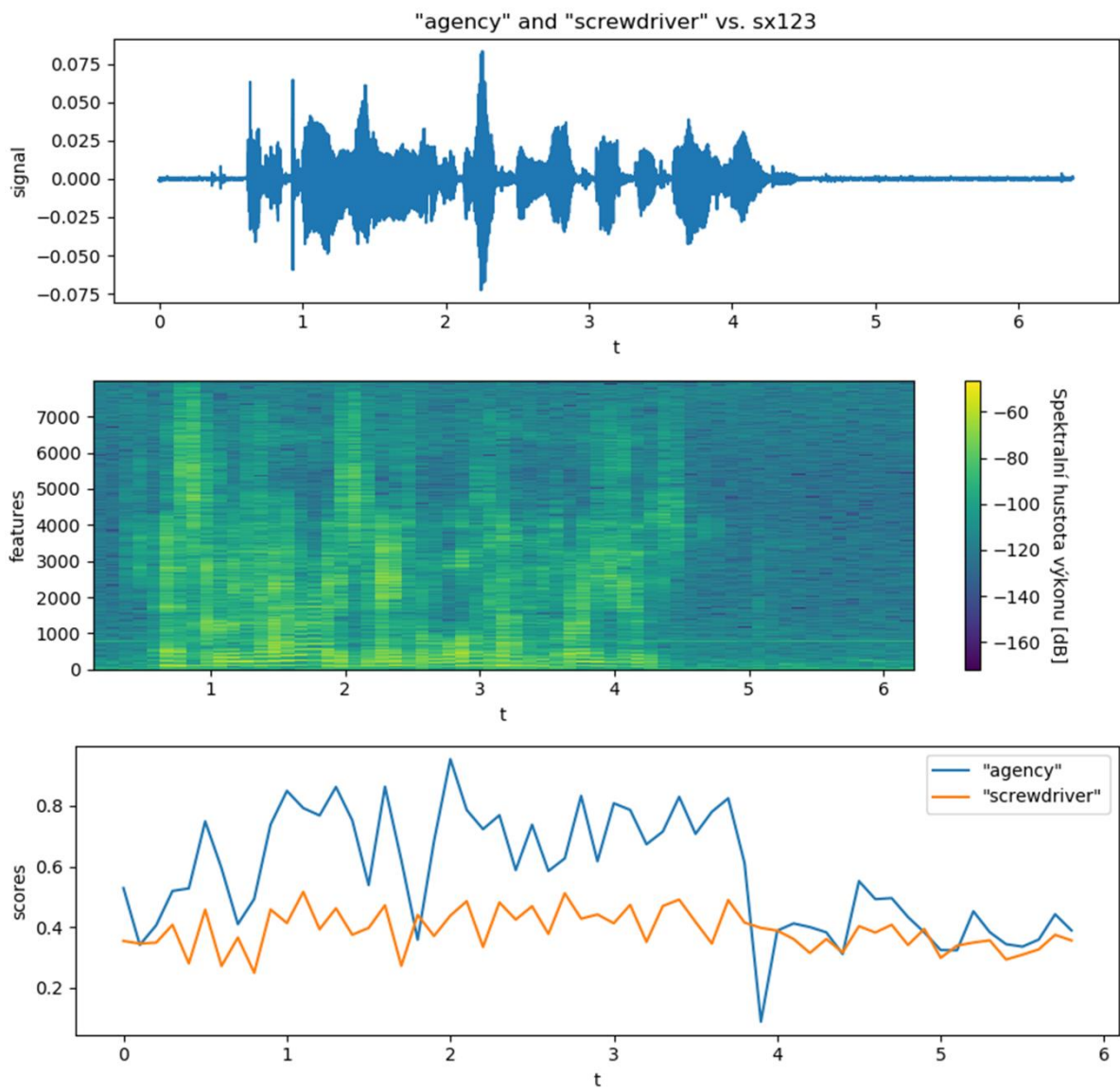


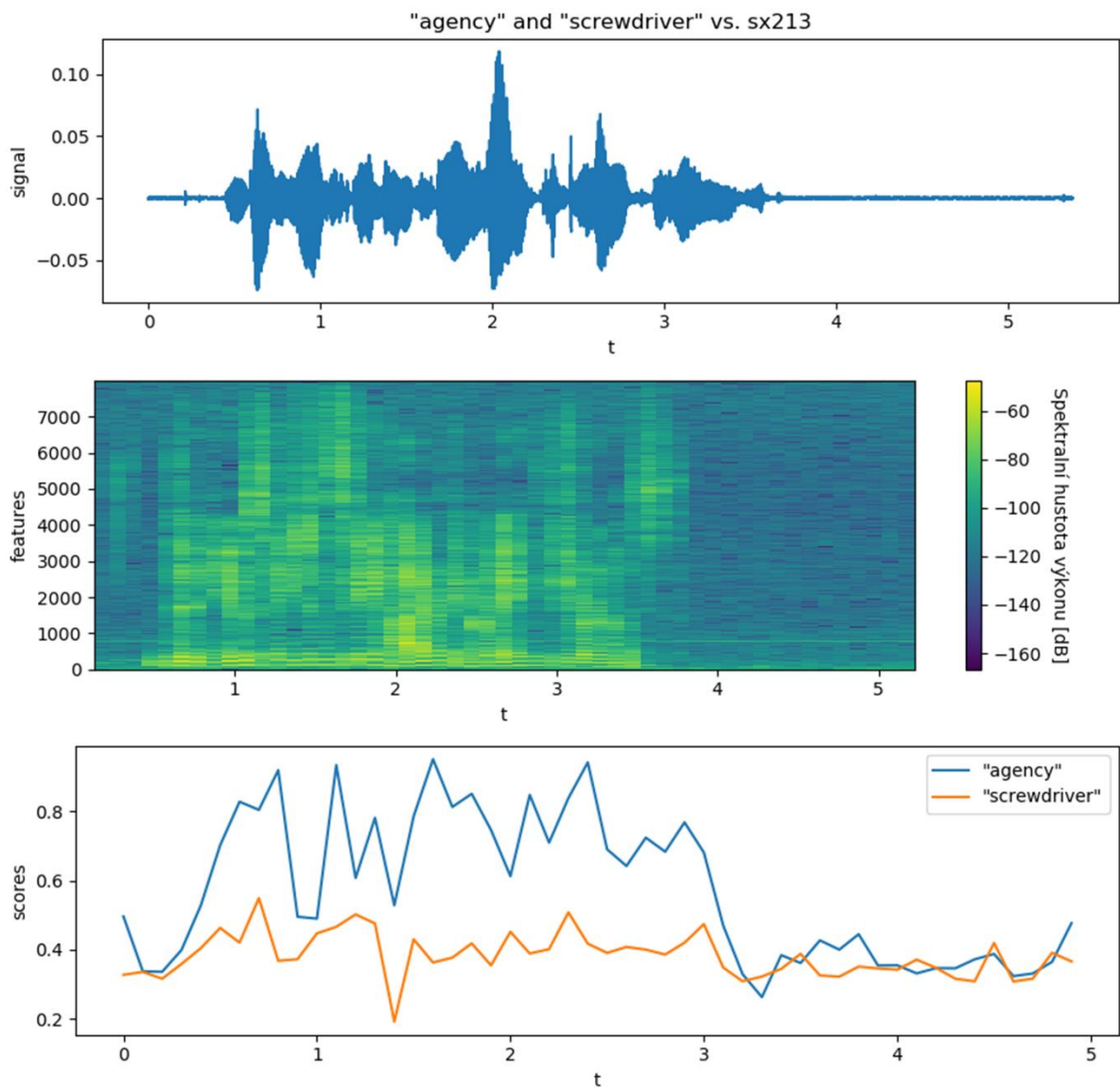


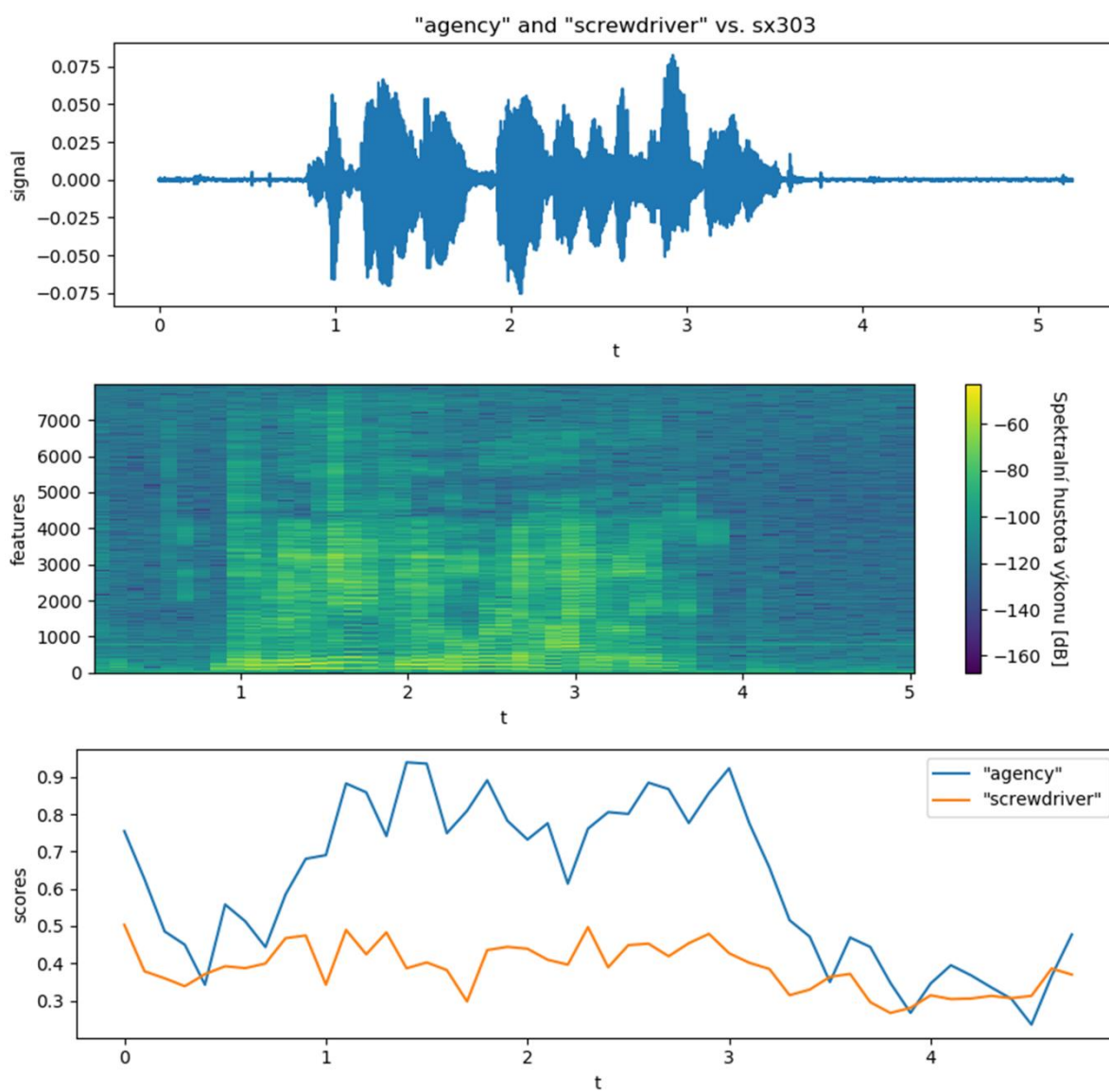


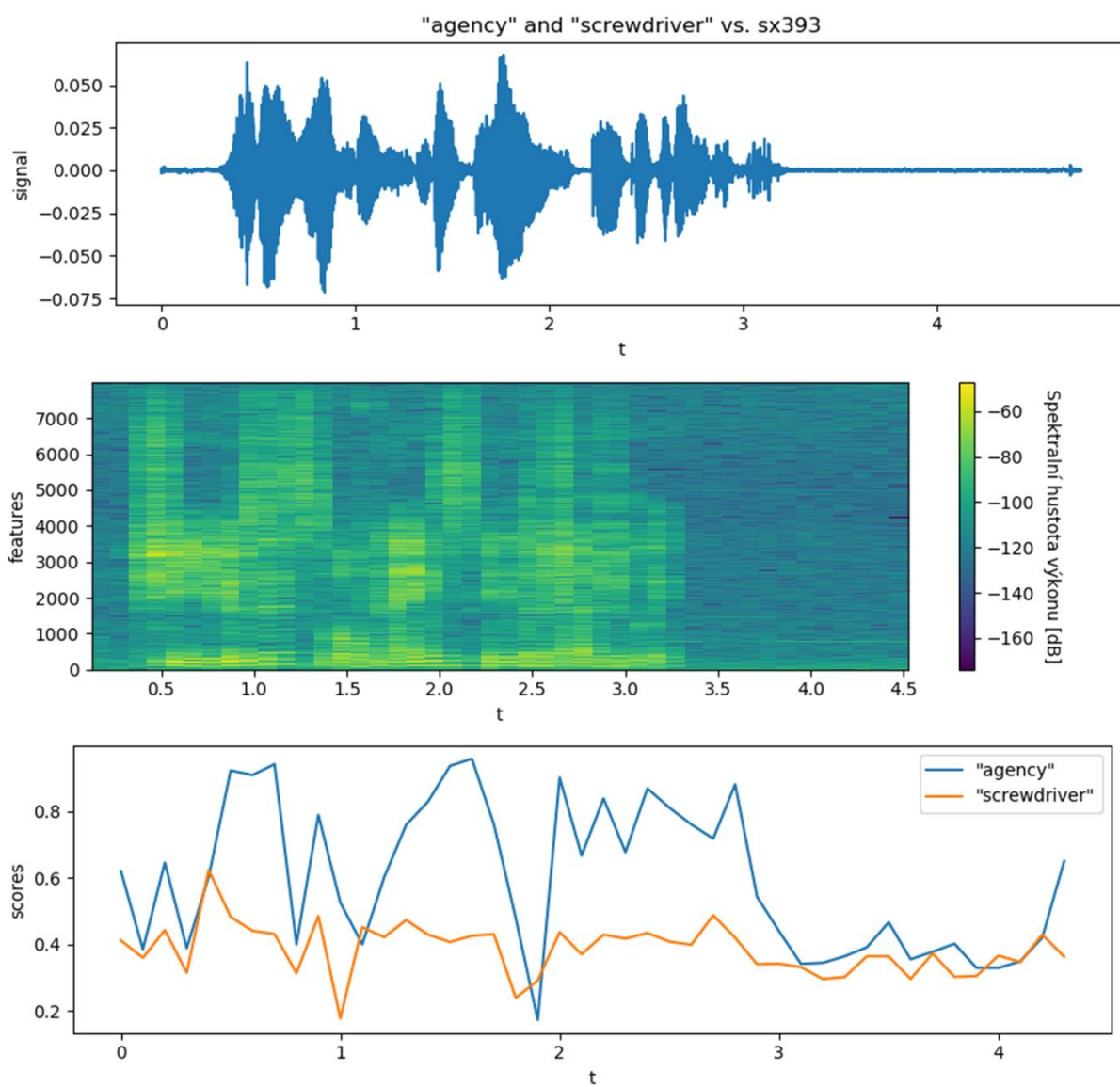












## Úkol 7:

Zda query na daném místě je nebo není bych určil tak, že určím práh 0.9 pro query 1 a práh 0.6 pro query 2, přiložím celé Q, tentokrát se všemi parametry do místa, kde je práh přesažen a vypočítal bych nové koeficienty korelace na celém úseku. Podle výsledků bych určil nový práh, který by již rozhodl, zda se v daném místě slovo nachází, či nikoliv. V mém hodnocení jsem testoval pouze prvních 10 parametrů Q, při přiložení všech parametrů (cca 60-70), by byl výsledek mnohem přesnější.

## Závěr

Můj pokus o detektor hodnotím jako velmi cennou zkušenost, i když výsledek není zcela uspokojivý. Výkyvy mezi query 1 a query 2 jsou znepokojivě velké, ačkoliv v úsecích, kde je klid, se hodnoty vcelku shodují. Je zcela možné, že se jedná o chybu ve výpočtu, nebo jsou výchyly dané jinou intonací a intenzitou hlasu při nahrávání každé query zvlášť. Celkově byl však projekt velmi přínosný a z mého pohledu i velmi dobře promyšlený.