ŠIAULIŲ UNIVERSITETAS
TECHNOLOGIJOS, FIZINIŲ IR BIOMEDICINOS MOKSLŲ FAKULTETAS
INŽINERIJOS KATEDRA

Greta Borcovaitė

KALBOS SIGNALŲ AKUSTINIŲ MODELIŲ, SKIRTŲ KALBĖTOJO ATPAŽINIMUI, TYRIMAS

Magistro darbas

Vadovas
prof. dr. G. Daunys

Šiauliai, 2017

ŠIAULIŲ UNIVERSITETAS

TECHNOLOGIJOS, FIZINIŲ IR BIOMEDICINOS MOKSLŲ FAKULTETAS

INŽINERIJOS KATEDRA

TVIRTINU
Katedros vedėja

_____

(parašas)
doc. dr. L. Kelpšienė

2017 m._____mėn.___d.

Greta Borcovaitė

KALBOS SIGNALŲ AKUSTINIŲ MODELIŲ, SKIRTŲ KALBĖTOJO ATPAŽINIMUI, TYRIMAS

Magistro darbas

Recenzentas

lekt. R. Zemblys

2017 06

Vadovas

prof. dr. G. Daunys

2017 06

Atliko

RM-15 gr. stud.

G. Borcovaitė

2017 06

TURINYS

SANTRAUKA

Borcovaitė G., Kalbos signalų akustinių modelių, skirtų kalbėtojo atpažinimui, tyrimas: Signalų technologijos magistro darbas / mokslinis vadovas prof. dr. G. Daunys; Šiaulių universitetas, Technologijos, fizinių ir biomedicinos mokslų fakultetas, Inžinerijos katedra, Šiauliai, 2017. –115 p.

Žmonės supranta matomą vaizdą, išgirstą garsą be papildomų mąstymo pastangų. Geba vienas kitą atpažinti – tam pakanka žvilgsnio ar ištarto žodžio. Tačiau technikos srityje net ir pats nesudėtingiausias automatizuotas objekto atpažinimas reikalauja daug pastangų. Dėl šios priežasties biometrijoje atliekama daug tyrimų, ieškoma naujų metodų, užtikrinančių greitesnį bei patikimesnį žmogaus tapatybės nustatymą.

Įvairioms balso technologijų užduotims atlikti plačiai naudojami atviro kodo paketai HTK, Kaldi ir MSR Identity Toolbox. Jais galima įvykdyti garso signalo tyrimą ir fonetinį išlyginimą, t. y. pagal duotą transkripciją garso signale sužymėti fonemų ribas.

Magistro baigiamajame darbe atliekamas balso biometrijos tyrimas. Darbo tikslas – ištirti kalbos signalų akustinius modelius, tinkančius kalbėtojų atpažinimui.

Kalbos signalų akustinių modelių sudarymui atliktas garso įrašų tyrimas, išskirti MFCC požymiai, nustatytos modelių komponenčių statistikos, sudaryti anglų, ispanų, italų, prancūzų, rusų ir vokiečių kalbų akustiniai modeliai taikant GMM ir k-vidurkių metodus, atliktas sudarytų kalbos signalų akustinių modelių patikrinimas.

Tyrimo metu išsiaiškinta, kad komponentės pasiskirsto įrašuose nevienodai. Įvertinant modelius išrinktos 6 dažniausiai pasitaikančios modelių komponentės. Be to nustatyta, kad dažniausiai pasitaikančios balso ir fono sričių komponentės yra skirtingos.

Statistinės analizės metu gauta, kad skirtinų kalbų įrašų logaritminiai tikėtinumai statistiškai reikšmingai nesiskiria, taikant tų kalbų to paties tipo modelį, o įrašų logaritminiai tikėtinumai nuo įrašų kalbų statistiškai reikšmingai nesiskiria, taikant anglų kalbos akustinius modelius. Taip pat nustatyta, kad ispanų ir anglų kalbų įrašų logaritminiai tikėtinumai labiausiai skiriasi. Padidinus įrašų imtį logaritminiai tikėtinumai statistiškai reikšmingai skiriasi anglų ir ispanų kalboms, taikant anglų kalbos akustinius modelius.

SUMMARY

Borcovaitė G., Research of speech signal acoustic models for speaker recognition. Master's thesis of Signal Technology / research advisor Assoe. dr. G. Daunys; Šiauliai university, Faculty of technology, physical and biomedical sciences, Department of Engineering, Šiauliai, 2017. –115 p.

Without much effort people can understand what they see, what other people say. One gaze or phrase is enough to recognize each other. However, in technical sphere even the easiest automatic object recognition is a hard task. A huge number of researches occur in biometrics for faster, more reliable and accurate human recognition.

HTK, Kaldi and MSR Identity Toolbox are widely used for various voice technology tasks. These open source packages could be apply for audio signal analysis and phonetic alignment i.e. for phoneme boundaries segmentation for given transcription.

The research of speech signal acoustic models for speaker recognition been described in this paper. The aim of this thesis – to investigate acoustic speech signal models suitable for speaker recognition.

In the analytical practical part voice records were investigate, MFCC features were extracted, statistics of acoustic models components were determined, GMM and k-mean acoustic speech signal models were trained for English, Spanish, Italian, French, Russian and German languages, investigation of created acoustic models was done.

Furthermore, investigation results has shown that components in records distributed differently. The six most common acoustic models components were chose. Moreover, it turned out that most common voice and background components are different.

Statistical analysis has shown that log-likelihoods are not statistically significant different for different language records when the same type and the same language acoustic models were applied. Besides, log-likelihoods are not statistically significant different for different language records when English acoustic models were used. Finally, log-likelihoods differ mostly in Spanish and English records. Increasing the number of English and Spanish records log-likelihoods are statistically significant different when English acoustic models are applied.

SUTRUMPINIMŲ SĄRAŠAS

DCF – sprendinio nuostolio funkcija (angl. Decision Cost Function).

DCT – diskretinė kosinuso transformacija (angl. Discrete Cosine Transform).

DFT – diskretinė Furjė transformacija (angl. Discrete Fourier Transform).

DNR – deoksiribonukleorūgštis (angl. Deoxyribonucleic Acid).

EER – vienodų klaidų reikšmė (angl. Equal Error Rate).

EM algoritmas – maksimalaus tikėtinumo algoritmas (angl. Expectation Maximization Algorithm)

FFT – greitoji Furjė transformacija (angl. Fast Fourier Transform).

GMM – Gauso klasterių modelis(angl. Gaussian Mixture Model).

LSTM – ilgos trumpalaikės atminties tinklas (angl. Long Short-Term Memory)

MFCC – Mel skalės kepstriniai koeficientai (angl. Mel-Frequency Cepstral Coefficient).

$\mathbb{N}$ – natūraliųjų skaičių aibė (angl. Set of Natural Numbers)

$\Re^d$ – d-matė realiųjų koordinačių erdvė (angl. Real Coordinate Space of d dimensions)

sup-GMM – prižiūrimas Gauso klasterių metodas (angl. supervised Gaussian Mixture Model).

TDNN – vėlinimo neuronų tinklas (angl. Time Delay Neural Network).

LENTELIŲ SĄRAŠAS

PAVEIKSLŲ SĄRAŠAS

ĮVADAS

Tyrimo aktualumas. Žmonės supranta matomą vaizdą ar girdimą garsą be jokių papildomų mąstymo pastangų, geba atpažinti vienas kitą iš trumpo žvilgsnio arba išgirsto žodžio. Iškyla klausimas, koks informacijos kiekis apdorojamas norint atlikti atpažinimo veiksmus. Technikos srityje net ir pats paprasčiausias automatizuotas objekto atpažinimas reikalauja nemažai pastangų ir resursų.

Biometrijoje individui identifikuoti atliekami tikslūs tam tikrų jo kūno parametrų matavimai. Pati biometrijos sąvoka apibūdinama kaip įvairūs technologiniai būdai, skirti asmens tapatybės nustatymui pagal žmogaus fiziologines ir, arba elgsenos savybes. Biometriniai duomenys, kuriais nustatoma žmogaus tapatybė yra [39]: DNR, ausies forma, akies rainelė, tinklainė, veidas, pirštų antspaudai, pirštų geometrija, delno geometrija, eisena, rankos geometrija, kvapas, rašysena, parašas, venos (esančios ant delno arba pirštų), balsas. Balso biometrija remiasi tokiomis asmens unikaliomis kalbos savybėmis kaip žmogaus vokalinio trakto fiziologijos ir specifinės kalbėsenos požymių išskyrimu, jų lyginimu kalbančiųjų skaičiaus, kalbėtojo atpažinimui, kalbų, dialekto ar akcento nustatymui.

Šių dienų kalbėtojo atpažinimo technologija plačiai taikoma visame pasaulyje. Bankai siūlo šią technologiją kaip greitą ir paprastą alternatyvą slaptažodžiams, pin numeriams, įsimintinoms datoms ar kortelių patvirtinimo numeriams – patvirtinimui užtenka trumpo pokalbio su atsakingų institucijų operatoriais [14]. Be to kalbančiojo tapatybės nustatymas taikomas įėjimo kontrolės punktuose, automobilių užrakto kontrolei, duomenų bazėms apsaugoti, kriminalistikoje arba atliekant teismo ekspertizes. Teismo ekspertizių metu asmens atpažinimas dažniausiai nepriklausomas nuo teksto, kadangi įprastai tiriamasis nelinkęs kalbėti, o ir garso įrašų sąlygos dažniausiai nebūna palankios: skirtingų asmens emocinių būsenų, triukšmo įtaka, garso įrašymo kanalų neatitikimo ir daug kitų, įrašo kokybę lemiančių veiksnių [3]. Nuo teksto nepriklausomam kalbėtojo atpažinimui gali būti naudojami MFCC požymiai ar GMM [19].

Nepaisant to, kad kalbėtojo atpažinimo tikslumas mažesnis už pirštų antspaudų, delno ar tinklainės skanavimą, balso biometrijos techninė įranga ženkliai pigesnė lyginant su kitų sričių identifikavimo sistemų įrenginiais – pakanka bet kokio garso įrašymo funkciją atliekančio įrenginio [26]. Tai gali būti diktofonas, išmanusis telefonas, planšetiniai, stacionarūs arba nešiojami kompiuteriai, muzikiniai grotuvai ir kt.

Tyrimo objektas – kalbos signalų akustiniai modeliai.

Darbo tikslas – ištirti kalbos signalų akustinius modelius, tinkančius kalbėtojų atpažinimui.

Uždaviniai:

- atlikti kalbos požymių ir akustinių modelių straipsnių apžvalgą;
- pasirinkti kalbos signalų akustinių modelių tyrimo metodiką;
- sukurti įvairių kalbų akustinius modelius, tinkančius kalbėtojų atpažinimui;
- atlikti kalbos signalų akustinių modelių tyrimą.

Tyrimo metodai:

- mokslinės literatūros analizė;
- mašininis mokymas;
- statistinė analizė.

Darbą sudaro santraukos lietuvių ir anglų kalbomis, įvadas, trys skyriai (straipsnių apžvalga, tyrimo metodika ir organizavimas, rezultatų analizė), išvados, literatūros sąrašas ir priedai.

Teorinėje dalyje aprašoma kalbos signalų akustinių modelių samprata, apibūdinami paslėptieji Markovo modeliai, aptariami kalbos požymiai, aprašoma teorinė neuronų tinklų apžvalga. Taip pat apžvelgiamos atviro kodo atpažinimo sistemos: HTK, Kaldi, MSR Identity Toolbox.

Praktinėje analitinėje darbo dalyje akustinių modelių sudarymui atliktas garso įrašų tyrimas, išskirti MFCC požymiai, sudaryti kalbos signalų akustiniai modeliai taikant GMM ir k-vidurkių klasterizavimo metodus, atliktas sukurtų akustinių modelių tyrimas.

Magistro darbo tema skaitytas pranešimas ŠU Technologijos, fizinių ir biomedicinos mokslų fakulteto organizuojamoje 12-oje mokslinėje konferencijoje „Studentų moksliniai darbai“.

1. STRAIPSNIŲ APŽVALGA: KALBOS POŽYMIAI IR AKUSTINIAI MODELIAI

1.1. Kalbos signalų akustiniai modeliai

Kalba – tai skirtingų garsų seka. Žmogaus smegenys geba klasifikuoti garsus į pagrindinius fonetinius vienetus, dar kitaip vadinamus fonemomis. Iš fonemų sekų sudarinėjami žodžiai. Žvelgiant iš atpažinimo pusės, žmogaus smegenys geba atpažinti kalbą bei kalbėtojus nepriklausomai nuo skirtingos aplinkos ar kalbėtojų, tačiau kompiuteriui tai nėra triviali užduotis [31].

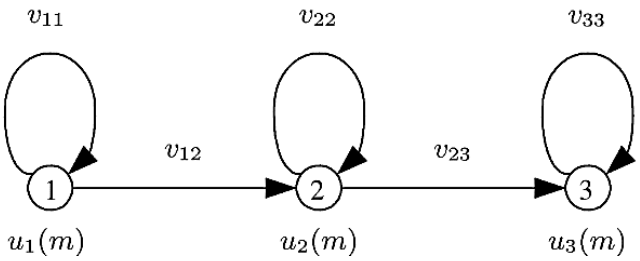
Šiuolaikinės technologijos sparčiai tobulėja, žengdamos į priekį kuria naujus būdus ir metodus, padedančius sunkiausias užduotis įvykdyti lengviau, tiksliau ir greičiau.

Kalbos signalų akustiniai modeliai – tai statistiniai modeliai, segmentuojantys tiriamus įrašus į modeliais aprašomas klases. Modeliai mokomi iš anksto paruoštais kelių valandų trukmės kalbos įrašais. Bendrinant modelį mokomoji medžiaga pateikiama skirtingo amžiaus ir lyčių asmenų. Toliau darbe apžvelgiami egzistuojantys kalbos signalų akustiniai modeliai ir kalbos požymiai.

1.2. Paslėptieji Markovo modeliai

Paslėptieji Markovo modeliai kalbos atpažinimui pritaikyti dar praėjusio amžiaus aštuntajame dešimtmetyje ir iki pat šių dienų yra vienas plačiausiai naudojamų kalbos atpažinimo metodų [8]. Dažniausiai šie modeliai naudojami ištisinei kalbai atpažinti, tačiau gali būti pritaikyti ir pavienių žodžių atpažinime.

Metodas remiasi prielaida, kad kalbos signalas gali būti gerai apibūdintas kaip atsitiktinis parametrinis procesas, o proceso parametrai gali būti įvertinti [12]. Paslėptųjų Markovo modelių metodu kalbos signalai gali būti modeliuojami paslėptaisiais Markovo modeliais (žr. 1.1 pav.).



1.1 pav. Trijų būsenų Markovo modelis

Šiuose modeliuose nagrinėjamas dvigubas atsitiktinis procesas: perėjimas iš vienos būsenos į kitą bei simbolių būsenoje generavimas. Dažniausiai naudojami pirmos eilės Markovo modeliai, kuriuose perėjimo tikimybė priklauso tik nuo ankstesnės būsenos. Paslėptuoju vadinamas pirmasis procesas, nes yra stebimas per antrąjį procesą.

Paslėptuosius Markovo modelius nusako tokie parametrai: modelio būsenų skaičius, skirtingų stebimų simbolių vienoje būsenoje skaičius, perėjimo tarp būsenų tikimybė, tam tikroje būsenoje stebimų simbolių tikimybės pasiskirstymas bei pradinis būsenos pasiskirstymas.

Pažymėkime N kaip modelio būsenų skaičių. Nors būsenos paslėptos, bet dažniausiai siejamos su tam tikra fizine reikšme. Bendruoju atveju, būsenos tarpusavyje susijusios taip, kad iš bet kurios būsenos būtų galima patekti į bet kurią kitą, dažnai išliekant toje pačioje būsenoje. Pažymėkime $S = S_1, S_2, \dots, S_N$ kaip atskiras būsenas, t – būsenos laiko momentą, q_t – būseną laiko momentu t.

Diskretaus žodyno dydis M – tai skaičius skirtingų stebimų simbolių vienoje būsenoje. Stebimi simboliai atitinka rizikinę modeliuojamos sistemos išėjime gaunamą reikšmę. Atskirus simbolius pažymėkime $V = v_1, v_2, \dots, v_M$.

Būsenos perėjimo tikimybė $A = a_{ij}$, kur $a_{ij} = P[q_{t+1} = S_j | q_t = S_i]$. Šiuo atveju turime, kad iš bet kurios būsenos galima patekti į bet kurią kitą būseną, kai $a_{ij} > 0, \forall\, i, j \,\epsilon\, \mathbb{N}$. Kitais atvejais $a_{ij} = 0$ vienai ir daugiau i ir j porų. Būsenoje j stebimų simbolių tikimybės pasiskirstymas $B = \{b_j(k)\}$. Čia $b_j(k) = P[v_k, \text{kai } t | q_t = S], 1 \le j \le N, 1 \le k \le M$.

Pradinis būsenų pasiskirstymas $\pi = \{\pi_i\}$, kur $q\pi_i = P[q_1 = S_i], 1 \le i \le N$.

Turint tinkamas N, M, A, B ir π reikšmes, paslėptasis Markovo modelis gali būti panaudojamas generuojant stebėjimų seką $O = O_1, O_2, \dots, O_T$, čia $\forall O_t \epsilon V$, o T – stebėjimų skaičius sekoje.

Būsenų seką generuojame algoritmu [23]:

1) pagal pradinį būsenų pasiskirstymą π parenkama pradinė būsena $q_1 = S_i$;
2) nustatoma $t = 1$;
3) pagal simbolių tikimybės pasiskirstymą būsenoje S_i, t. y. $b_i(k)$, parenkama $O_t = v_k$;
4) pagal būsenų perėjimo tikimybės pasirinkimą būsenoje S_i, t. y. a_{ij}, pereinama į būseną $q_{t+1} = S_j$;
5) nustatoma $t = t + 1$ ir jei $t < T$, tai grįžtama į 3 žingsnį, priešingu atveju procedūra baigiama.

Procedūra gali būti naudojama ne tik stebėjimų sekų generavimui, bet ir modeliavimui, kai duotoji seka sugeneruota tinkamu paslėptuoju Markovo modeliu.

Vadinasi, visiškai paslėptojo Markovo modelio specifikacijai reikia dviejų modelio parametrų N ir M, stebimų simbolių bei trijų tikimybinių matricų A, B, π. Trumpumo dėlei modelis gali būti išreiškiamas trimis parametrais $\lambda = (A, B, \pi)$.

1.2.1. Trys pagrindiniai uždaviniai

Sakykime turime W kalbos pavyzdžių, kurių atpažinimui naudojamas paslėptasis Markovo modelio metodas. Pirmasis žingsnis – žodyno sukūrimas. Kiekvienam iš W pavyzdžių sukuriame modelį λ. Modelio būsenų ir stebėjimų būsenoje skaičių lemia pasirinktas modelio tipas, modeliuojamas kalbos pavyzdys, priimtos pradinės prielaidos. Tikimybiniai modelio parametrai A, B, π nustatomi taikant įvertinimo procedūras iš mokymui pateiktų pavyzdžių.

Nagrinėjant nežinomą pavyzdį atliekama signalo analizė, kurios metu gaunama stebėjimų seka O. Atpažintuoju pavyzdžiu traktuojamas etaloninis pavyzdys, t. y. kuris modelis geriausiai atitinka nagrinėjamą stebėjimų seką. Tikėtinumu įvertinant modelio atitikimą stebėjimų sekai, skelbiamas etalonas $Z = \arg \max_{0<k<W} P(O|\lambda_k)$.

Siekiant naudingai praktikoje taikyti paslėptuosius Markovo modelius sprendžiami uždaviniai:

1) tikėtinumo $P(O|\lambda)$ apskaičiavimas – tai įvertinimo uždavinys, kaip apskaičiuoti tikimybę, kad stebima seka buvo sugeneruota duoto modelio, ar kaip gerai duotas modelis atitinka duotą seką;
2) optimalios būsenų sekos parinkimas esant užduotai stebėjimų sekai O ir modeliui λ – tai uždavinys, kuriame bandoma atidengti paslėptojo modelio dalis;
3) parametrų, maksimizuojančių tikėtinumą $P(O|\lambda)$, parinkimas – uždavinys, kuriame bandoma taip optimizuoti modelio parametrus, kad jie kuo geriau aprašytų, kaip gauta duotoji stebėjimų seka.

Tiesiogiai skaičiuojant tikėtinumo reikšmę $P(O|\lambda)$ susiduriama su milžiniškomis skaičiavimo apimtimis. Problemai išspręsti pasiūlyti rekurentiniai skaičiavimo būdai [24]. Juose apibrėžiami tarpiniai kintamieji – daliniais tikėtinumais organizuojamas iteracinis tikėtinumų skaičiavimas, kiekvieną dabartinę tarpinio kintamojo reikšmę išreiškiant per buvusią reikšmę.

Optimalios būsenų sekos parinkimo uždavinys kur kas sudėtingesnis nei tikėtinumo skaičiavimas stebėjimų sekos ir modelio atžvilgiu, kadangi yra keletas galimų sekos optimalumo kriterijų bei tiesioginis visų galimų sekų perrinkimas ir optimalios sekos išrinkimas reiškia milžinišką skaičiavimų kiekį. Dažniausiai naudojamas tikėtiniausios sekos kriterijus, maksimizuojantis sekos tikėtinumą. Sekai nustatyti pasiūlytas Viterbi algoritmas [41], realizuojamas dinaminiu programavimu. Kaip efektyvesnė alternatyva pasiūlytas modifikuotas Viterbi algoritmas [24], kurio realizavimui reikalingas mažesnis operacijų skaičius.

Paskutinysis uždavinys sudėtingiausias, nes neįmanoma analitiškai gauti modelio λ parametrų, maksimizuojančių stebėjimų sekos tikėtinumą $P(O|\lambda)$. Lokalus tikėtinumo maksimumas gali būti pasiektas Baum-Welch algoritmu, kuris iš esmės yra dalinis vidurkio maksimizavimo atvejis [27].

Tiriant kalbos signalus paslėptųjų Markovo modelių metodu, randami fundamentalūs kalbos elementai [38], tokie kaip: fonemos, nuo konteksto priklausomos fonemos, žodžiai, žodžių sekos. Dažniausiai naudojamos keleto žodžių frazės bei kiekvienam žodžiui kuriami modeliai, kurie kombinuojami sakinio lygyje, atkreipiant dėmesį į sakinio lygio gramatiką.

Paslėptojo Markovo modelio metodo privalumai: didelis atpažinimo tikslumas nepriklausantis nuo kalbėtojo, kalbos pavyzdžių modeliavimas susietomis būsenomis leidžia nesunkiai į akustinį apdorojimą įjungti lingvistinio apdorojimo elementus.

Trūkumai: sudarant etaloninių kalbos pavyzdžių modelius dažnai į juos įtraukiama papildoma lingvistinė informacija. Tokie modeliai tampa priklausomais nuo tikslinės kalbos ir skirtingoms kalboms skiriasi, dėl to šiuo atveju pritaikant atpažinimo sistemą kitai kalbai, mokymo kitos kalbos pavydžiais neužteks – teks kurti naujus modelius; metodo prigimtis reikalauja nemažo mokymo duomenų kiekio; atpažinimo tikslumo praradimo.

1.3. Kalbos požymiai

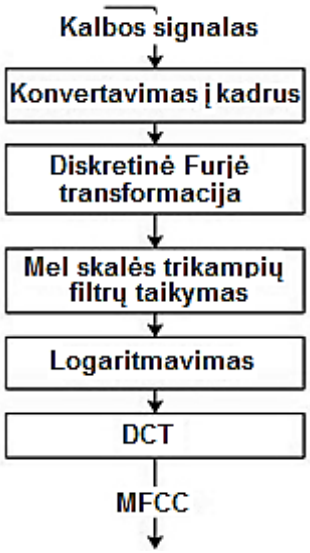
Kalbos požymių išskyrimo etapas yra vienas svarbiausių momentų kalbėtojo atpažinimo procese, nuo kurio tiesiogiai priklauso identifikavimo tikslumas. Todėl geras požymių parinkimas lemia atpažinimo rezultatus.

1.3.1. MFCC

Siekiant pagerinti kalbos atpažinimo kokybę esant pašaliniams triukšmams, pradėtos kurti naujos požymių sistemos. Sistemos kurtos atkreipiant dėmesį į garso signalo apdorojimo modelį, pavyzdžiui, žmogaus ausies vidinės sraigės darbą. Modeliai taikyti netiesinių dažnių skalės požymių skaičiavimui, kartu ir Mel skalės kepstro koeficientų skaičiavimui (MFCC).

Šiuolaikinėse kalbos atpažinimo sistemose plačiai naudojami MFCC [14]. Koeficientų apskaičiavimui atkreipiamas dėmesys į žmogaus garso suvokimo ir signalo struktūros požymius, t. y. ausis jautresnė dažnių pokyčiams žemuose dažniuose. Mel analizės etapai nurodomi 1.2 paveiksle.

Pirmajame etape kalbos signalas padalinamas į pakankamai mažus (dažniausiai 20 ms ilgio) signalo kadrus, kurie padauginami iš lango funkcijos. Praktikoje dažniausiai naudojama Hammingo lango funkcija. Kitame žingsnyje skaičiuojama kiekvieno kadro diskretinė Furjė transformacija (DFT). Toliau signalo spektras dauginamas iš Mel skalės trikampių funkcijų koeficientų.



1.2 pav. Mel analizės etapai

Sakykime, kiekvienas filtras aprašytas svorių funkcija arba charakteristika *f(i, k)*, tada kiekvieno filtro išėjime energija

$$S(m,i) = \sum_{k=0}^{N-1} |DFT(m,k)| f(i,k) \tag{1.1}$$

čia i – filtro indeksas, $1 \le i \le Q$, k – kepstro atskaitos indeksas, m – kadro numeris, $1 \le m \le M$, M – kadrų skaičius signale, Q – sveikas teigiamas skaičius.

Logaritmuojant $S(m,i)$ ir taikant diskretinę kosinuso transformaciją (DCT) gauname:

$$c_m(n) = \sum_{k=0}^{K} \cos\left[n(i-\frac{1}{2})\frac{\pi}{k}\right] \lg S(m,i), \tag{1.2}$$

čia $c_m(n)$ – MFCC, K – filtrų rinkinių skaičius, k – DFT spektro atskaitos indeksas.

Tyrinėjant kalbos signalus dažniausiai skaičiuojama apie 10–20 MFCC. Papildomai nustatomi dinaminiai kepstro koeficientai [21]:

$$\Delta c_i(n) = \frac{\sum_{k=-K}^{K} k c_i(n+k)}{\sum_{k=-K}^{K} k^2}, \tag{1.3}$$

čia K – dinaminis langas, $c_i(n)$ – i-tasis kepstro koeficientas, gautas n-tajam kadrui.

Be to buvo įrodyta, kad MFCC požymių kokybė prastėja esant triukšmingiems garso įrašams, skirtingiems įrašinėjimo įrenginiams, nes požymiai gali būti sugadinti įvairių triukšmų bei iškraipymų. Dėl to pradėtas taikyti MFCC normavimas. MFCC normavimo žingsniai:

1) randamos MFCC reikšmės $c_m(n)$ taikant (1.2) lygybe;
2) apskaičiuojamas visų kadrų vidurkis: $\mu = \frac{1}{M}\sum_{m=1}^{M} c_m(n)$;

3) iš kiekvienos MFCC reikšmės atimamas rastas vidurkis μ: $\Delta_m = c_m(n) - \mu$;

4) apskaičiuojamas visų kadrų standartinis nuokrypis: $s = \sqrt{\frac{1}{M}\sum_{m=1}^{M}(\Delta_m)^2}$;

5) MFCC dalijama iš standartinio nuokrypio s, kaip palyginimui su 3-iojo žingsnio rezultatu: $\frac{c_m(n)}{s}$.

Taigi, po normavimo kalbos signale panaikinami triukšmai ir iškraipymai.

Straipsnyje [38] atliktas nuo teksto priklausomo kalbėtojo atpažinimo eksperimentas, kurio metu taikytos skirtingos MFCC modifikacijos. Iš pradžių, keičiant filtrų skaičių bandyta išsiaiškinti, kuriuo atveju kalbėtojo atpažinimo tikslumas didžiausias. Gauti rezultatai vaizduojami 1.1 lentelėje.

1.1 lentelė. Kalbėtojo atpažinimo rezultatai taikant MFCC su skirtingų filtrų skaičiumi

Filtrų skaičius	12	22	32	42
Efektyvumas	65%	75%	85%	80%

Nustatyta, kad atpažinimo tikslumas didžiausias taikant 32 filtrus. Toliau tyrimas tęstas taikant skirtingas langų funkcijas: Hanningo ir stačiakampio (žr. 1.2 lentelė).

1.2 lentelė. Kalbėtojo atpažinimo rezultatai taikant MFCC su skirtingomis langų funkcijomis

Lango tipas taikant 32 filtrus	**Efektyvumas**
Hanningo	75%
Stačiakampio formos	55%

Gauta, kad efektyvumas didžiausias taikant 32 filtrus ir Hanningo lango funkciją – nuo teksto priklausomo kalbėtojo atpažinimo tikslumas 75%.

Nors ir šiuolaikinės technologijos tobulinamos sparčiai, buvo nustatyta, kad rišlios kalbos atpažinimas ir tikslumo gerinimas sietinas su naujų požymių paieškomis. Be to požymiai plačiai taikomi atliekant lyginamąsias analizes, t. y. kiek procentų rezultatas pagerintas lyginant su MFCC.

1.3.2. Statistiniai požymiai, panaudojant GMM

Gauso klasterių modelis (GMM) – pasisekęs automatinis balso atpažinimo metodas [28]. Metodu vertinamos požymių tikimybės. GMM yra tinkamas variantas modeliuojant tikimybinius skirstinius, kadangi metodu gali būti aproksimuota bet kokia tolydi tankio funkcija.

Vienmačiu atveju Gauso skirstinio tankio funkcija apibrėžiama funkcija:

$$f(x|\mu,\sigma^2) = \frac{1}{\sqrt{2\pi\sigma}}\exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right). \quad (1.4)$$

Čia μ, σ^2 – vidurkis bei dispersija atitinkamai, x – atsitiktinio kintamojo reikšmė.

Daugiamačiu atveju būsenos tankio funkcija charakterizuojama išraiška:

$$\begin{aligned} b_j(x_t) &= \sum_{m=1}^{M} c_{jm} N(x_t, \mu_{jm}, \Sigma_{jm}) \\ &= \sum_{m=1}^{M} c_{jm} \frac{1}{\sqrt{2\pi|\Sigma_{jm}|}} \exp(-\frac{1}{2}(x_t - \mu_{jm})^T \Sigma_{jm}{}^{-1}(x_t - \mu_{jm})). \end{aligned} \tag{1.5}$$

Čia $b_j(x_t)$ – būsenos tankio funkcija, $c_{jm}, \mu_{jm}, \Sigma_{jm}$ – svoriai, vidurkiai ir kovariacijos komponentės j būsenoje m atitinkamai, t – laiko indeksai, M – Gausianų skaičius.

Normaliojo skirstinio parametrai

$$\begin{aligned} \Upsilon'_t(j,m) &= \frac{P(x_{1:T}, s_t = j, m_t = m|\Theta)}{P(x_{1:T}|\Theta)} \\ &= \frac{\sum_i \alpha_{t-1}(i) a_{ij} c_{jm} N(x_t, \mu_{jm}, \Sigma_{jm}) \beta_t(j)}{\sum_{k=1}^{N_a} \alpha_T(k)} \end{aligned} \tag{1.6}$$

Čia $\Upsilon'_t(j,m)$ – komponentės j būsenoje m būvimo tikimybė.

Nauji, M-tajame žingsnyje rasti Baum-Welch algoritmo parametrai nustatomi taikant išraiškas:

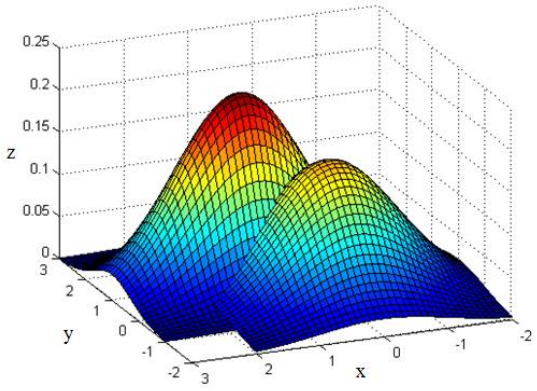
$$\hat{c}_{jm} = \frac{\sum_{t=1}^{T} \Upsilon'_t(j,m)}{\sum_{t=1}^{T} \sum_{m'=1}^{M} \Upsilon'_t(j,m')}, \tag{1.7}$$

$$\hat{\mu}_{jm} = \frac{\sum_{t=1}^{T} \Upsilon'_t(j,m) x_t}{\sum_{t=1}^{T} \Upsilon'_t(j,m)}, \tag{1.8}$$

$$\hat{\Sigma}_{jm} = \frac{\sum_{t=1}^{T} \Upsilon'_t(j,m)(x_t - \hat{\mu}_{jm})(x_t - \hat{\mu}_{jm})^T}{\sum_{t=1}^{T} \Upsilon'_t(j,m)}. \tag{1.9}$$

Čia $\hat{c}_{jm}, \hat{\mu}_{jm}, \hat{\Sigma}_{jm}$ – komponentės j būsenoje m svorių, vidurkių ir kovariacijų įverčiai atitinkamai.

GMM modelis su 2 požymių komponentėmis pateikiamas 1.3 paveiksle [36].



1.3 pav. GMM modelis su 2 požymių komponentėmis

Toliau paaiškinama, kaip gaunami modelio parametrai taikant Baum-Welch algoritmą. Algoritmo veikimo principas pagrįstas didžiausio tikėtinumo metodu. Parametrų didžiausio tikėtinumo įvertinių apskaičiavimas sprendžiamas skaitiniais metodais. Vienas tokių – maksimalaus tikėtinumo (EM) algoritmas, kuris vykdomas dviem etapais: tikėtinumų nustatymo, tikėtinumų maksimizavimo.

Modelio parametrų apskaičiavimui vykdomas Baum-Welch algoritmas [30]:

1) kiekvienam parametrų vektoriui ar matricai (1.7), (1.8) ir (1.9) lygybėmis išskiriama atmintis;
2) skaičiuojamos kiekvienos komponentės *j* būsenoje m tiesioginio ir atbulinio sklidimo tikimybės;
3) kiekvienai komponentei *j* būsenoje m tikimybė $\Upsilon_t'(j,m)$ ir nagrinėjamas stebėjimų vektorius x_t naudojami tos būsenos m išskirtos atminties patikslinimui;
4) galutinės išskirtos atminties reikšmės naudojamos naujų parametrų reikšmių apskaičiavimui;
5) jei šio etapo bendros tikimybės reikšmė didesnė už praėjusio etapo – skaičiavimai baigiami; priešingu atveju, naujosioms parametrų reikšmėms veiksmai kartojami nuo pradžių (nuo 1 etapo).

Baum-Welch algoritmas naudojamas Markovo modelio parametrų įvertinimui. Kai paslėptieji Markovo modeliai naudojami kaip modelio topologija, atliekant pradinius skaičiavimus bei nustatant parametrų skaičių, iškyla keletas svarbių klausimų. GMM atveju pabrėžiama, kad metodas gali sumodeliuoti bet kokį skirstinį su pakankamu normaliųjų klasterių skaičiumi, tačiau tai kompromisas tarp mokymo duomenų kiekio ir gautų tikimybinių skirstinių komponenčių skaičiaus. Kita vertus, siekiant sumažinti parametrų skaičių požymiai gali būti papildyti požymių transformacijų vidurkiais.

Straipsnyje [20] pasiūlyta kalbėtojo atpažinimui taikyti GMM drauge su MFCC. Atliekant eksperimentą parinktas 8 kHz dažnis, kalbos požymiai gauti tiriant 30 ms kadrus su 20 ms gretimų kadrų persidengimais. GMM parametrai apskaičiuoti taikant didžiausio tikėtinumo metodą. Straipsnyje gauti rezultatai – atliekant 16 įrašų testavimą gautas 87,5% atpažinimo tikslumas.

Taigi, GMM naudojamas vertinant požymių tikimybes. Metodas gali būti taikomas lygiagrečiai su kitais kalbėtojo atpažinimo metodais. Atliekant kalbėtojo atpažinimo užduotį GMM parametrai nustatomi taikant didžiausio tikėtinumo metodą.

1.3.3. Didžiausio tikėtinumo metodas

Didžiausio tikėtinumo metodas skirtas statistinio modelio parametrams įvertinti. Apskritai fiksuotam duomenų rinkiniui, taikant šį metodą, suteikiama galimybė gauti modelio parametrų rinkinių reikšmes, su kuriomis tikėtinumo funkcija maksimizuojama. Šiuo metodu nustatomos tokios parametrų reikšmės, su kuriomis gaunami rezultatai tampa tikėtiniausi duotajam modeliui [43].

Kuo didesnė imtis, tuo tikėtiniau, jog didžiausio tikėtinumo metodu gauti parametrų įverčiai skirsis mažai nuo tikrųjų parametro reikšmių. Bendruoju atveju įvertiniai [8]: pagrįsti – konverguoja pagal tikimybę į nežinomo ir vertinamo parametro reikšmę, asimptotiškai normalieji, efektyvūs – turi mažiausią dispersiją tarp visų galimų nežinomo parametro įvertinių.

Sakykime, atsitiktinis vektorius $X \in \Re^d$ aprašomas tikimybiniu tankiu $p(x|\theta)$, čia $\theta \in \Re^d$ – nežinomų parametrų vektorius. Duota X nepriklausomi vienodai pasiskirstę atsitiktiniai dydžiai $X^1, X^2, \dots, X^K$, su tankio funkcija $p(x|\theta)$. Atsitiktinio dydžio tankio funkcijų sandauga vadinama tikėtinumo funkcija ir apibrėžiama išraiška:

$$L(\theta) = \prod_{i=1}^{K} p\left(X^i|\theta\right). \tag{1.10}$$

Didžiausio tikėtinumo įverčiais parenkamos tokios parametrų θ reikšmės, su kuriomis tikėtinumo funkcija $L(\theta)$ įgyja didžiausią reikšmę, o parametrų θ įvertinys vadinamas didžiausio tikėtinumo įvertiniu. Siekiant palengvinti skaičiavimus tikėtinumo funkcija $L(\theta)$ logaritmuojama.

Diferencijuojant tikėtinumo funkciją, funkcijos $\log L(\theta)$ maksimumai ieškomi tokiu būdu:

1) surandamos dalinės išvestinės $\frac{dL(\theta)}{d\theta_j}$, $j = 1, 2, \dots, n$;
2) prilyginus išvestines nuliui sprendžiama lygčių sistema

$$\begin{cases} \dfrac{dL(\theta)}{d\theta_1} = 0, \\ \dfrac{dL(\theta)}{d\theta_2} = 0, \\ \dots \\ \dfrac{dL(\theta)}{d\theta_n} = 0. \end{cases} \tag{1.11}$$

   su n lygčių ir n nežinomųjų, kur dažniausiai turi vienintelį sprendinį;
3) naudojantis $X^1, X^2, \dots, X^K$ atsitiktinių dydžių skirstinio didžiausio tikėtinumo įvertinių asimptotiniu normališkumu, apskaičiuojami didžiausio tikėtinumo įvertinių pasikliautinieji intervalai.

Parametrų didžiausio tikėtinumo įvertinių apskaičiavimas sprendžiamas skaitiniais metodais. Vienas tokių – EM algoritmas [32]. Apskritai didžiausio tikėtinumo parametrų nustatymas – iteracinis procesas, kuriam parenkamos pradinės įvertinių reikšmės ir reikiamų iteracijų skaičius, kol greta esančių įverčių reikšmių skirtumas tampa pakankamai mažas.

Straipsnyje [22] taikomas didžiausio tikėtinumo metodas kalbėtojo atpažinime. Čia didžiausio tikėtinumo svorių parametrų įverčiai su laisvaisiais parametrais laikomi tiesinio programavimo problema. Eksperimento metu išsiaiškinta, kad patikimumo kokybę lemia mikrofonų bei sesijos (laiko) pokyčiai, t. y. nurodytieji pokyčiai statistiškai reikšmingi.

Vadinasi, taikant didžiausio tikėtinumo metodą, įvertinami statistinio modelio parametrai. Šio metodo privalumai: apskaičiuojami regresinių modelių parametrų įverčiai, o esant didelėms stebėjimo imtims, apskaičiuoti įverčiai turi pageidaujamas savybes. Trūkumas: skaičiavimų sudėtingumas. Taip pat, trūkumu laikoma tai, kad būtina žinoti priklausomojo kintamojo tikimybių pasiskirstymą.

1.3.4. Filtrų rinkinių galingumai

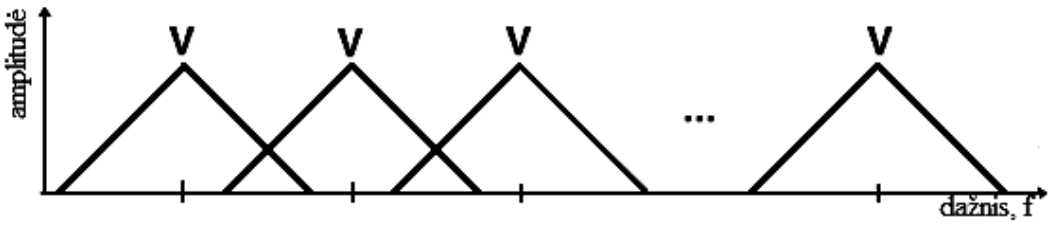
Kalbos signalai apdorojami skaitmeniniais filtrais. Filtrų rinkiniai – vieni plačiausiai naudojamų atpažįstant kalbos signalus. Filtrų rinkiniais skaitmeninis kalbos signalas $s(t)$ praleidžiamas per Q filtrų rinkinių bloką, padengiantį dominantį signalo dažnių diapazoną. Dažniausiai filtrų skaičius tenkina sąlygą: $8 \leq Q \leq 32$.

Filtrų bloko taikymui reikalingos parengiamosios operacijos: triukšmo pašalinimas, signalo gerinimas bei signalo spektro išlyginimas.

Triukšmo pašalinimas gali eliminuoti kalbos signalo dedamąją, atsiradusią dėl pašalinio aplinkos triukšmo bei neturinčią nieko bendro su kalbos signalu. Signalo gerinimas siejamas su formančių pikų aštrumo didinimu, o signalo spektro išlyginimas eliminuoja būdingą signalo spektro nuolydį.

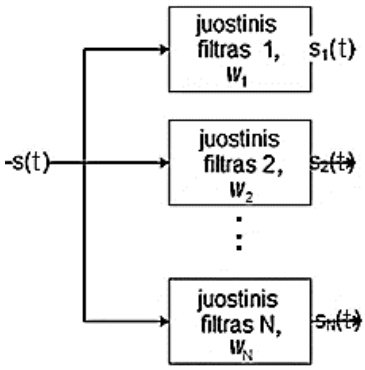
Filtrų bloko išėjime, po viso proceso, atliekamos užbaigiamosios operacijos, kurių tikslas – išvalyti požymius nuo filtrų bloko poveikio, kad šie kuo efektyviau atstovautų kalbos signalo spektrinė informaciją ir maksimaliai padidintų sėkmingo kalbos signalo atpažinimo galimybę.

Pavieniai filtrai dažniausiai persidengia dažnių srityje (žr. 1.4 pav.).



1.4 pav. Filtrų rinkinio bloko amplitudinė charakteristika

Tarkime, i-tojo filtrų rinkinio išėjimas – kalbos signalo $s(t)$ trumpalaikio spektro atvaizdavimas su centriniu dažniu w_i pralaidumo rinkinyje i-tuoju laiko momentu, kur $\leq i \leq N$. Tada toks filtrų rinkinio blokas gali būti realizuojamas lygiagrečiais filtrais (žr. 1.5 pav.).



1.5 pav. Lygiagrečių filtrų rinkinio blokas

Sakykime, kalbos signalui $s(t)$, praėjus pro filtrų rinkinio bloką, išėjimuose turime [19]:

$$s_i(t) = s(t) * h_i(t) = \sum_{k=0}^{M_{i-1}} h_i(k)s(t-k). \tag{1.12}$$

Čia $h_i(k)$ pažymi i-tojo filtrų rinkinio impulsinę reakciją, kurios ilgis M_i.

Kiekviena signalo $s_i(t)$ atskaita keliama kvadratu. Signalo spektras pasislenka į žemų dažnių sritį, bet jame taip pat lieka aukštų dažnių komponentės. Aukštų dažnių komponenčių pašalinimui naudojamas žemų dažnių filtras, kurio juostos plotis 20-30 Hz, dėl to po žemų dažnių filtro seka išretinimo blokas, kurio išėjime žemų dažnių filtro signalas diskretizuotas 40-60 Hz dažniu. Signalo amplitudės transformacija gali būti atliekama po išretinimo.

Elementariausias filtrų rinkinio blokas sudarytas iš vienodo pralaidumo filtrų rinkinio juostos pločio. Jei N – filtrų rinkinio filtrų skaičius, reikalingas perdengti visą kalbos signalo dažnių diapazoną, tai f_i – i-tojo filtrų rinkinio normuotas centrinis dažnis:

$$f_i = \frac{F_i}{F_5}, i = 1,2,\dots,Q, \text{ kur } F_i = \frac{iF_5}{2N}, Q < N. \tag{1.13}$$

Čia $F_i - i$-tojo filtrų rinkinio dažnis, $Q -$ tikrasis filtrų rinkinio filtrų skaičius, $Q < N$, o $F_s -$ signalo diskretizavimo dažnis.

i-tojo filtrų rinkinio juostos plotis turi tenkinti sąlygą: $b_i \geq \frac{F_5}{2N}$. Jei išraiškoje vietoje nelygybės turime lygybę – filtrų dažnių juostos nepersidengia. Idealiomis sąlygomis persidengimo gali ir nebūti, tačiau realiomis tai neišvengiama.

Kitas filtrų rinkinio išdėstymo būdas – logaritminėje dažnių skalėje. Tokiu atveju filtrų rinkinių aibei Q centriniai dažniai F_i ir filtrų rinkinių juostų pločiai $b_i \leq i \leq Q$ tenkina sąlygas:

$$b_i = C, \tag{1.14}$$

$$b_i = \alpha b_{i-1}, i = 2,3,\dots,Q, \tag{1.15}$$

$$F_i = F_1 + \sum_{j=1}^{i-1} b_j + \frac{(b_i - b_1)}{2}. \quad (1.16)$$

Čia $C -$ pirmo filtro juostos plotis ir $F_1 -$ centrinis dažnis, $\alpha -$ logaritminis juostos pločio prieaugio daugiklis. Reikšmė $\alpha = 2$ atitinka gretimų filtrų išdėstymą pagal oktavą, $\alpha = \frac{4}{3}$ atitinka $\frac{1}{3}$ oktavos filtrų išdėstymą [28].

Nepaisant to, kad vienodo juostų pločio filtrų blokai naudojami kalbos atpažinime, tačiau paplitę nevienodo filtrų juostų pločio blokai – jais sumažinami skaičiavimai, o kalbos signalo spektras apibūdinamas analogiškai, kaip tai suvokia žmogus.

1.3.5. Tiesinės prognozės koeficientai

Neretai praktikoje susiduriama su prognozavimo uždaviniais, kuriuose tenka numatyti dominančios sistemos elgseną būsimais laiko momentais. Sakykime, žinomas sistemos išėjimo signalas esamu laiko momentu $t - 1$. Reikia prognozuoti, koks bus išėjimas kitu laiko momentu t.

Tiesinės prognozės metodu yra įvertinamos formantės, atskiriant jas nuo kalbą generuojančio šaltinio, lemiančio kalbos garsumą ir toną. Dažnių sritis kalbos signalo spektre, turinti didžiausią energiją, vadinama formante. Pagal tariamo garsą ir dažnių juostą gali būti stebimos 3-5 formantės.

Tarkime, turime tokią baigtinės energijos duomenų seką $y(t)$, kai $-\infty < t < \infty$. Šiomis sekomis galima sukurti sistemą, kuri prognozuotų realios sistemos elgseną, būtų ribotos impulsinės reakcijos, tiesinė ir nekintanti laike. Dėl to reikalingas prognozuojančios sistemos modelis:

$$y'(t) = -a_1 y(t-1) - a_2 y(t-2) - \cdots - a_p y(t-p). \quad (1.17)$$

Kadangi energijos duomenų seka $y(t)$ žinoma – galima rasti momentinį nuokrypį $e(t)$:

$$e(t) = y(t) - y'(t). \quad (1.18)$$

Žinant momentinio nuokrypio reikšmę randama nuokrypių kvadratų suma:

$$E_N = \sum_{t=-\infty}^{\infty} e^2(t) = \sum_{t=-\infty}^{\infty} [y(t) - y'(t)]^2 = \sum_{t=-\infty}^{\infty} \left[y(t) + \sum_{k=1}^{p} a_k y(t-k) \right]^2. \quad (1.19)$$

Sprendžiant lygčių sistemą randami $\frac{\partial E_N}{\partial a_k} = 0$, kai $k = 1,2, \dots, p$.

Toliau aiškinamas koreliacinis metodas. Koreliacinio metodo tiesinės prognozės modelio parametrai $a_p(m), m = 1,2, \dots, p$, įvertinami taikant lygybes [2][13]:

$$E_0 = r(0), \quad (1.20)$$

$$a_i^{(i)} = k_i = \frac{r(i) - \sum_{j=1}^{i-1} a_j^{(i-1)} r(|i-j|)}{E^{(i-1)}}, \qquad i = 1,2,\dots,p, \tag{1.21}$$

$$a_i^{(i)} = a_j^{(i-1)} + k_i a_{i-j}^{(i-1)}, \qquad 1 \le j \le i-1, \tag{1.22}$$

$$E^{(i)} = \left(1 - k_i^2\right) E^{(i-1)}, \tag{1.23}$$

$$a_0^p = 1 \text{ (pagal apibrėžimą).} \tag{1.24}$$

Čia $r(p)$ – autokoreliacinė funkcija, $r(0)$ – signalo energija, $E^{(i)}$ – prognozės klaidos energija.

Tačiau (1.17) įrašius į (1.18) turime

$$y(t) + a_1 y(t-1) + a_2 y(t-2) + \cdots + a_p y(t-p) = e(t). \tag{1.23}$$

Į signalą $y(t)$ galima žiūrėti kaip į tiesinės nekintančios laike sistemos, kurios įėjimo signalas $e(t)$, išėjimą. Toliau pereinama prie išraiškos

$$Y(z) + a_1 z^{-1} Y(z) + a_2 z^{-2} Y(z) + \cdots + a_p z^{-p} Y(z) = E(z), \tag{1.24}$$

$$Y(z)(a_1 z^{-1} + a_2 z^{-2} + \cdots + a_p z^{-p}) = E(z). \tag{1.25}$$

Tada, sistemos funkcija

$$H(z) = \frac{Y(z)}{E(z)} = \frac{1}{1 + a_1 z^{-1} + a_2 z^{-2} + \cdots + a_p z^{-p}}. \tag{1.26}$$

Įvertinus sistemos funkciją $H(z)$ ant vienetinio apskritimo gaunama dažninė charakteristika

$$H\left(e^{j\omega}\right) = H(z)|_{z=e^{j\omega}} = \frac{1}{1 + a_1 e^{-j\omega} + a_2 e^{-2j\omega} + \cdots + a_p e^{-pj\omega}}. \tag{1.27}$$

Formule (1.27) nustatomi amplitudės spektras ir energijos tankio spektras. Dėl to įvedamas keitinys:

$$e^{jx} = \cos x + j \sin x, \tag{1.28}$$

$$e^{-jx} = \cos x - j \sin x. \tag{1.29}$$

Tada dažninė charakteristika

$$\begin{aligned} H\left(e^{j\omega}\right) &= \frac{1}{1 + a_1 e^{-j\omega} + \cdots + a_p e^{-pj\omega}} \\ &= \frac{1}{1 + a_1(\cos\omega - j\sin\omega) + a_2(\cos 2\omega - j\sin 2\omega) + \cdots + a_p(\cos p\omega - j\sin p\omega)} \\ &= \frac{1}{\left(1 + a_1\cos\omega + a_2\cos 2\omega + \cdots + a_p\cos p\omega\right) + j(-a_1\cos\omega - a_2\cos 2\omega - \cdots - a_p\cos p\omega)} \\ &= \frac{1}{A + jB}. \end{aligned}$$

Apibrėžiamas energijos tankio spektras [2]:

$$|H\left(e^{j\omega}\right)|^2 = H\left(e^{j\omega}\right) \cdot H\left(e^{j\omega}\right)^* = \frac{1}{A + jB} \cdot \frac{1}{A - jB} = \frac{1}{A^2 + B^2}. \tag{1.30}$$

Čia $H\left(e^{j\omega}\right)^*$ – kompleksiškai jungtinė šaknis.

Apibrėžiamas amplitudės spektras

$$\left|H\left(e^{j\omega}\right)\right| = \sqrt{|H(e^{j\omega})|^2} = \frac{1}{\sqrt{A^2 + B^2}}. \tag{1.31}$$

Be to tiesinės prognozės parametrų skaičiavimo metu nustatomi atspindžio koeficientai $k_m, m = 1,2, \dots, p$ arba dar kitaip vadinamieji dalinės koreliacijos koeficientais. Iš atspindžio koeficientų nustatoma, ar tiesinės prognozės modelis stabilus.

Taikant atspindžio koeficientus galima apskaičiuoti vamzdžių plotų santykius:

$$g_m = \ln\left(\frac{1 - k_m}{1 + k_m}\right), \qquad m = 1,2, \dots, p. \tag{1.32}$$

Tarp tiesinės prognozės parametrų, logaritminių plotų santykių bei atspindžio koeficientų egzistuoja abipusė vienareikšmė priklausomybė. Atspindžio koeficientais rekurentiškai apskaičiuojami metodo parametrai [2][18]:

$$a_i^{(i)} = k_i, \qquad 1 \le i \le p, \tag{1.33}$$

$$a_j^{(i)} = a_j^{(i-1)} k_i a_{i-j}^{(i-1)}, \qquad 1 \le j \le i - 1, \tag{1.34}$$

$$a_j = a_j^{(p)}. \tag{1.35}$$

Atspindžio koeficientai gali būti skaičiuojami iš metodo parametrų taikant rekurentines lygtis

$$k_i = a_i^{(i)}, \qquad 1 \le i \le p\,. \tag{1.36}$$

$$a_j^{(i-1)} = \frac{a_j^{(i)} + a_i^{(i)} a_{i-j}^{(i)}}{1 - k_i^2}, \qquad 1 \le j \le i - 1. \tag{1.37}$$

Atspindžio koeficientus galima apskaičiuoti ir iš logaritminių plotų santykių formulių. Dėl to abi formulės puses parrašomos laipsnyje e, vietoj kintamojo m imami i, $1 \le i \le p$, $e^{g_i} = e^{\ln\left(\frac{1-k_i}{1+k_i}\right)}$. Tada iš logaritminių savybių turime $e^{g_i} = \frac{1-k_i}{1+k_i}$. Atliekami pakeitimai $e^{g_i} = -\frac{k_i+1-2}{k_i+1}$, $e^{g_i} = -\left(1 - \frac{2}{k_i+1}\right)$, $1 + e^{g_i} = \frac{2}{k_i+1}$, $k_i + 1 = \frac{2}{1+e^{g_i}}$, $k_i = \frac{2}{1+e^{g_i}} - 1$. Pertvarkius gaunama formulė $k_i = \frac{1-e^{g_i}}{1+e^{g_i}}$, pagal kurią iš atspindžio koeficientų apskaičiuojamas logaritminių plotų santykis.

Priklausomai nuo kalbėtojo atpažinime taikomos klasifikavimo taisyklės požymiai gali būti naudojami ne tik kaip parametrai, bet ir atspindžio koeficientai ar logaritminių plotų santykiai.

Taigi, tiesinė prognozė – vienas galingiausių akustinių kalbos signalų analizės metodų, grąžinančių labai gerus kalbos bei kalbos kompresijos kokybės parametrus. Šiuo metodu gaunami kalbą aprašantys parametrai apskaičiuojami greitai, o jų skaičius nėra didelis.

Tiesinė prognozė plačiai naudojama kalbėtojo atpažinimo sferoje, tačiau pasiekta riba ir rezultatai nebegerėja. Dėl to, siekiant sumažinti klaidų skaičių, pradėti kurti ir taikyti nauji metodai.

1.4. Teorinė neuronų tinklų apžvalga

Dirbtiniai neuronų tinklai kalbėtojų atpažinimui pradėti taikyti dar dvidešimto amžiaus devintajame dešimtmetyje ir tai yra vienas naujausių metodų. Šios srities pasiekimai suvesti knygoje [15]. Toliau apžvelgiami konvoliuciniai tinklai ir LSTM bei jų taikymo galimybės kalbėtojo atpažinime.

1.4.1. Konvoliuciniai neuronų tinklai

Konvoliuciniai neuronų tinklai – tai neuronų tinklo sinapsių svorių modifikacijos, atsirandančios iš patobulinto atgalinio sklidimo algoritmo, kurio veikimo principas pagrįstas gradiento ir globalaus mokymo algoritmais, kurie filtruodami ir klasifikuodami pasiekia daugiasluoksnio tinklo struktūrą, t. y. atlieka požymių suradimo operaciją su poslinkio, mastelio ir deformacijos paklaida.

Konvoliuciniai neuronų tinklai gali būti taikomi daugybėje skirtingų sričių, tyrinėjančių balsą, tekstą arba vaizdus. Šie tinklai yra vienas iš moderniausių būdų atpažįstant dvimačius šablonus, klasifikuojant įvairius vaizdus ar sakinius, veido taškus, kalbėtojus ir kt.

Konvoliucinius neuronų tinklus sudarantys sluoksniai [6]: konvoliucinių, mažinimo ir visiškai sujungtųjų. Konvoliuciniai sluoksniai – tai pagrindinė konvoliucinio neuronų tinklo dalis. Kiekvienas konvoliucinis sluoksnis sudarytas iš dvimačių neuronų plokštumų, dar vadinamų požymių žemėlapiais. Kiekvienas požymių žemėlapio neuronas sujungtas su neuronų grupe iš stačiakampio tinklo, esančio ankstesniame sluoksnyje. Visų konvoliucinio sluoksnio neuronų svoriai vienodi, todėl kiekvienas neuronas iš to paties požymių žemėlapio dalijasi tokiais pat svoriais.

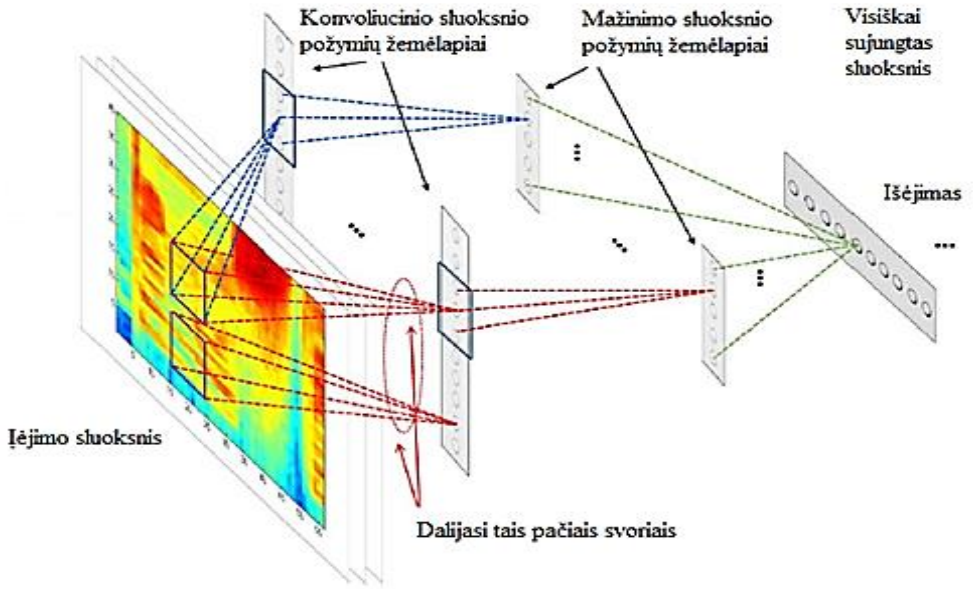
Kiekviename konvoliuciniame sluoksnyje gali būti daugiau nei vienas požymių žemėlapių. Naudojantis potencialiai skirtingais filtrais, iš kiekvienų ankstesnių sluoksnių paimamos visos reikšmės į kiekvieną kito sluoksnio požymių žemėlapį.

Po kiekvieno konvoliucinio sluoksnio dažniausiai naudojami mažinimo sluoksniai, sumažinantys požymių žemėlapį. Šie sluoksniai sumažina ankstesnių sluoksnių požymių žemėlapių neuronų skaičių, sujungdami gretimus neuronus į vieną. Mažinimo sluoksnių požymių žemėlapių skaičius toks pat kaip ir ankstesniuose sluoksniuose. Kiekvienas požymių žemėlapis jungiamas tik su atitinkamu ankstesnio sluoksnio požymių žemėlapiu. Egzistuoja keletas būdų mažinimui atlikti. Vienas jų – vidurkio arba maksimumo taikymas.

Po visų mažinimo ir konvoliucinių sluoksnių neuronų tinkle vykdomas aukšto lygio samprotavimas per visiškai sujungtus sluoksnius. Šiais sluoksniais klasifikuojami duomenys. Visiškai sujungti

sluoksniai paima neuronus iš ankstesnių sluoksnių ir sujungia su kiekvienu ankstesniame sluoksnyje esančiu neuronu. Po visiškai sujungto sluoksnio daugiau konvoliucinių sluoksnių būti negali.

Išvardytieji sluoksniai išdėstyti taip, kad rezultatas iš ankstesniojo sluoksnio būtų perduodamas į kitą. Konvoliucinio neuronų tinklo pavyzdys pateikiamas 1.6 paveiksle [1].



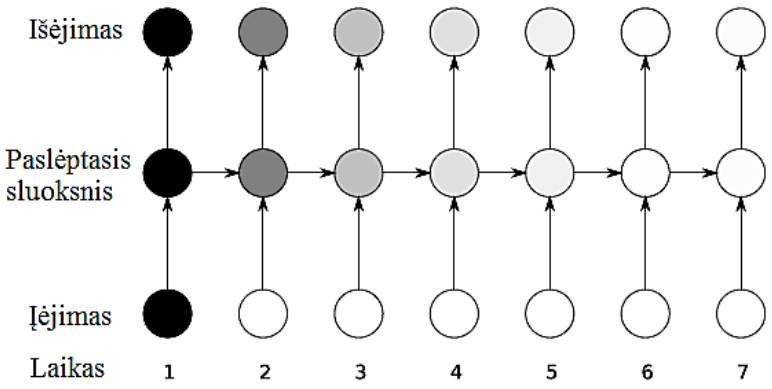
1.6 pav. Konvoliucinio neuronų tinklo pavyzdys

Straipsnyje [5] pristatytas kalbos atpažinimo tyrimas konvoliuciniais neuronų tinklais. Šiam tikslui pasiekti taikyta 12 sluoksnių: pirmieji 2 sluoksniai – mažinimo, paskutinieji sluoksniai – visiškai sujungti, o likusieji – konvoliuciniai sluoksniai. Įprastai neuronų tinklai, kurių sandara pagrįsta MFCC ar galios spektru, neturi pirmųjų mažinimo sluoksnių. Gauti rezultatai – tiek taikant galios spektrą, tiek neapdorotus požymius rezultatai prastesni nei taikant MFCC.

Apibendrinus galima teigti, konvoliucinius neuronų tinklus sudaro trys pagrindinės sluoksnių rūšys: konvoliuciniai, mažinimo bei visiškai sujungtieji. Taikant konvoliucinius neuronų tinklus gali būti sumodeliuoti įvairūs būdai, tinkantys kalbėtojo, sakinių ar vaizdų atpažinimui.

1.4.2. LSTM

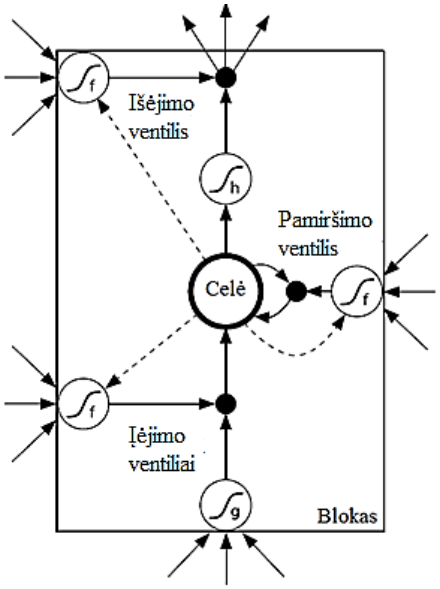
Ilgos trumpalaikės atminties tinklo (LSTM) architektūrą sudaro aibė rekurentiškai sujungtų potinklių – atminties blokų [10]. Kiekvieną bloką sudaro viena ar daugiau tarpusavyje sujungtų atminties celių ir trys dauginimo įtaisai: įėjimų, išėjimų ir pamiršimo ventilių išėjimų. Įtaisai vykdo rašymo, skaitymo arba atnaujinimo operacijas. LSTM tinklo veikimas laike pateikiamas 1.7 paveiksle.



1.7 pav. LSTM veikimas laike

Atspalviai tinklo mazguose nurodo jautrumą. Tai vadinama nykstančio gradiento problema, t. y. mašininio mokymo sunkumais, atsirandančiais neuronų tinklo mokymui, taikant atbulinio sklidimo arba gradientu paremtais mokymo metodus. Jautrumas laikui bėgant slopsta, nes nauji įėjimo signalai perrašo paslėptojo sluoksnio reikšmes ir tinklas pamiršta pirmąjį įėjimą. Tokiu būdu pašalinama nykstančio gradiento problema.

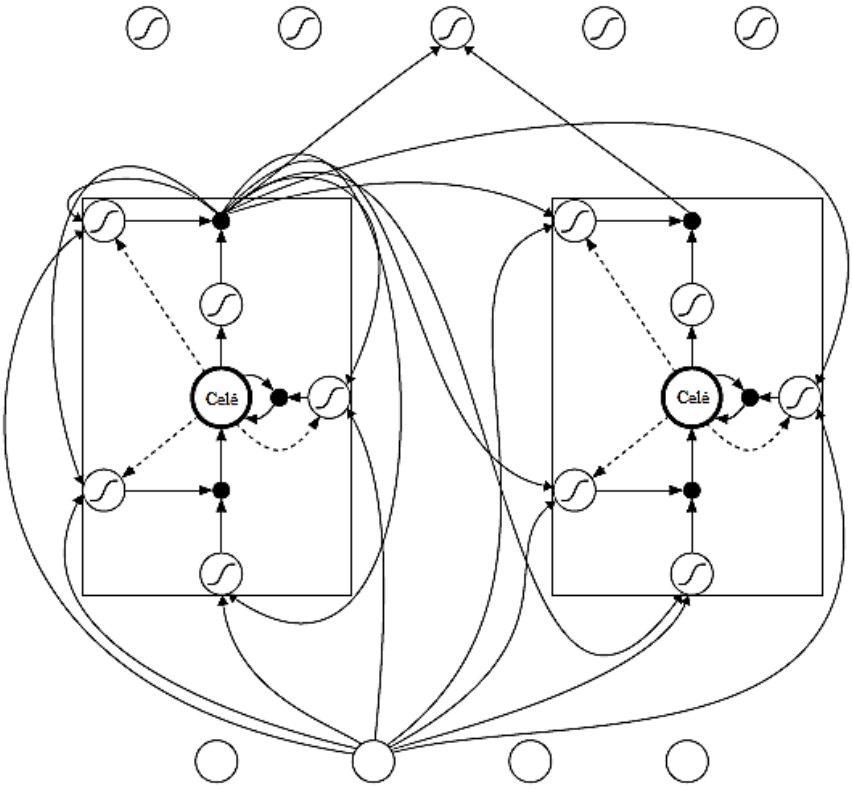
LSTM atminties blokas su viena cele vaizduojamas 1.8 paveiksle.



1.8 pav. LSTM atminties blokas su viena cele

Apskritai LSTM tinklas analogiškas standartiniam rekurentiniam neuronų tinklui, išskyrus tai, kad sumavimo operacijos paslėptajame sluoksnyje keičiamos atminties blokais. Išėjimo sluoksnis gali būti toks pat kaip ir įprastinio rekurentinio neuronų tinklo.

Dauginimo įtaisai LSTM atminties celėms leidžia saugoti ir gauti informaciją – tokiu būdu sušvelninama nykstančio gradiento problema. Dviejų celių LSTM struktūra vaizduojama 1.9 paveiksle.



1.9 pav. Dviejų celių LSTM tinklas

Matematiškai LSTM celės gali būti apibūdinamos žemiau pateiktomis iteracinėmis išraiškomis laiko momentais $t = 1, 2, \dots, T$ [15]:

$$i_t = \sigma\left(W^{(xi)}x_t + W^{(hi)}h_{t-1} + W^{(ci)}c_{t-1} + b^{(i)}\right), \quad (1.38)$$

$$f_t = \sigma\left(W^{(xf)}x_t + W^{(hf)}h_{t-1} + W^{(cf)}c_{t-1} + b^{(f)}\right), \quad (1.39)$$

$$c_t = f_t \bullet c_{t-1} + i_t \bullet tanh\left(w^{(xc)}x_t + w^{(hc)}h_{t-1} + b^{(c)}\right), \quad (1.40)$$

$$o_t = \sigma\left(W^{(xo)}x_t + W^{(ho)}h_{t-1} + W^{(co)}c_t + b^{(o)}\right), \quad (1.41)$$

$$h_t = o_t \bullet tanh(c_t), \quad (1.42)$$

čia i_t, f_t, c_t, o_t ir h_t – tos pačios dimensijos vektoriai, saugantys įėjimo ventilio, pamiršimo ventilio, aktyvavimo celės, išėjimo ventilio bei paslėptojo sluoksnio vektorius, atitinkamai, $\sigma()$ – logaritminė sigmoido funkcija, W – skirtingus ventilius jungiančios svorių matricos, b – poslinkio vektoriai. Visos svorių matricos yra pilnos, išskyrus diagonalią matricą $W^{(ci)}$.

LSTM taip pat taikomas sprendžiant įvairias realių žodžių problemas, tokias kaip muzikos generavimas, kokybės gerinimas, kalbėtojo ar rankraščio atpažinimas. Tinklo privalumai pasireiškia tose srityse, kur sprendžiamos problemos, reikalaujančios didelės apimties kontekstinės informacijos.

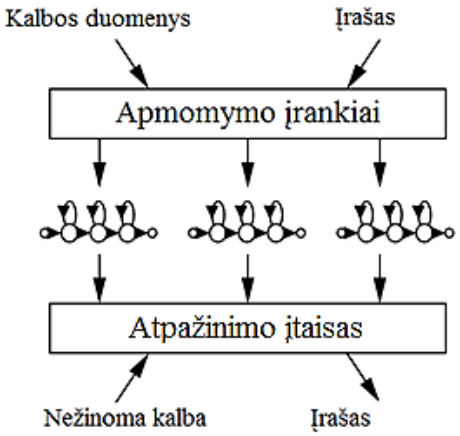
1.5. Atviro kodo atpažinimo sistemos

Įvairioms balso technologijų užduotims atlikti plačiai naudojami atviro kodo paketai HTK [14], Kaldi [26] ir MSR Identity Toolbox [33]. Jais galima įvykdyti garso signalo tyrimą ir fonetinį išlygi-nimą, t. y. pagal duotą transkripciją garso signale sužymėti fonemų ribas.

1.5.1. HTK

HTK programinis paketas – vienas iš plačiausiai naudojamų atpažinimo įrankių. Sistema su-kurta S. Young Kembridžo (Cambridge) universitete dar 1989 m. ir tobulinta nuo pirmosios versijos *Version 1* iki pat šių dienų naujausios *Version 3.5 beta*. Gerinant sistemą, prie S. Young prisijungė P. Woodland, G. Evermann, M. Gales.

Paketas skirtas realizuoti šnekos apdorojimo priemones, kurių veikimo principas pagrįstas pas-lėptaisiais Markovo modeliais. Apdorojimo mechanizmas pateikiamas 1.10 paveiksle [14].



1.10 pav. HTK apdorojimo mechanizmo schema

Taigi, apdorojimui taikomi mokymo ir atpažinimo įrankiai. Mokymo įrankiai naudojami apytiksliam paslėptųjų Markovo modelių rinkinio parametrų apskaičiavimui, taikant mokymo medžiagą (garso įrašus) ir su ja susijusias transkripcijas. Nežinomi ar neatpažinti fragmentai transkribuojami HTK kalbos atpažinimo įrankiais.

Siekiant HTK paketu vykdyti atpažinimą, kiekvienam nurodytam etapui taikoma atskira HTK sistemos rinkinio programa. Visos komandos vykdomos per eilutės komandas ir papildomus doku-mentus. Pagrindiniai darbo etapai: žodyno sudarymas, požymių išskyrimas, mokymas bei testavimas.

Projektuojant sistemą svarbu nustatyti, kokios fonemų kombinacijos leidžiamos, todėl būtinas žodyno sudarymas. Po šio etapo seka požymių paieška. Įprastai atpažinimo sistemose naudojami ne patys įrašai, o jų požymiai, todėl būtini įrašų aplankai. HTK sistemoje požymių išskyrimui naudojama Hcopy, kuris pagal nurodytus parametrus apdoroja kalbos signalą bei išskiria požymio vektorius.

HTK dokumento antraštė –12 baitų ilgio. Antraštėje saugomi tokie duomenys: *nSamples* – imčių skaičius (4 baitai), *sampPeriod* – imties periodas (4 baitai), *sampSize* – imties dydis (2 baitai) ir *parmKind* – imties tipas (2 baitai).

Kiekvieno iš galimų požymių parametrų tipą sudaro 6 bitų kodas. Pagrindiniai požymių parametrų tipai [15]: LPC – tiesinės prognozės koeficientai, LPREFC – tiesinės prognozės atspindžio koeficientai, LPCEPSTRA – tiesinės prognozės kepstro koeficientai, LPDELCEP – tiesinės prognozės kepstras bei delta koeficientas, IREFC – tiesinės prognozės atspindžio koeficientai, išsaugoti 16 bitų tikslumu, MFCC, PLP – percepcinės tiesinės prognozės koeficientai ir kt.

Atviro kodo fonemų atpažinimo sistemos HTK parametrai: WAVEFORM – skaliarinės imtys, FBANK – prijungti (log) filtrų rinkinio parametrai, MELSPEC – tiesiniai filtrų rinkinio parametrai, USER – vartotojo apibrėžti parametrai ir kt.

Vadinasi, HTK sistema gali būti randami požymiai: kalbos signalo analizės, tiesinės prognozės koeficientai, filtrų rinkiniai, kepstro požymiai, percepcinės tiesinės prognozės, energijos matmenys, delta, pagreičio ir kiti. Visi šie požymiai plačiau apibūdinti HTK knygoje [14].

Modelių mokymas įgyvendinamas Baum-Welch algoritmu, kuris aktyvuojamas paleidžiant programą HRest. Mokymo metu nurodomi pagrindiniai paslėptųjų Markovo modelių parametrai.

Modelių failai – tai specialaus formato aplankalai, kuriuose surašomos tam tikro modelio perėjimų bei išėjimų tikimybės. Išėjimų tikimybės modeliuojamos Gauso skirstiniais, nes fonemų atpažinimo sistema operuoja tik tolydžiais paslėptaisiais Markovo modeliais. Tai reiškia, kad rezultatų tikimybės aprašomos vidurkiais ir standartiniais nuokrypiais.

Toliau vykdomas testavimas. Testuojant patikrinama, kaip gerai vykdomas atpažinimas. Patikrinimą rekomenduojama vykdyti kitais įrašais. Testavimas vykdomas HVite programa.

Nemažai mokslininkų dirba su HTK sistema iki pat šių dienų. Darbe [29] apibūdinamas kalbėtojo atpažinimo sistemos veikimo principas pagrįstas HTK paketo bei MFCC požymiais. Tirti 10 kalbėtojų garso įrašai. Kiekvienas kalbėtojas paprašytas pasirinkti po vieną žodį, kaip slaptažodį, iš tam tikros Indų kalbos ir pakartoti jį 20 kartų. Tokiu būdu iš gautų 20 pakartojimų: 15 taikyti mokymui ir 5 testavimui. Autorių sukurtą duomenų bazę sudarė 10 kalbėtojų, 150 mokymo pavyzdžių bei 50 testavimo pavyzdžių. Jų gauti rezultatai pateikiami 1.3 lentelėje [29].

1.3 lentelė. Kalbėtojo atpažinimo rezultatai

Kalbos	Mokymo tikslumas, %	Testavimo tikslumas, %
Telugų	100,00	99,57
Hindi	100,00	99,14
Urdų	97,14	89,71
Orijų	99,18	94,28
Bendgalų	98,44	94,28
Tamil	100,00	99,41
Kanadų	99,60	99,23
Malajalių	98,94	97,80
Marati	99,87	98,40
Anglų	100,00	100,00
Vidurkis	**99,32**	**97,18**

Gautas testavimo tikslumas mažesnis nei mokymo. Tačiau sunku įvertinti tokius rezultatus, nes nėra nurodyti frazių ilgiai.

Kalbėtojo atpažinime įvairūs metodai bei sistemos gali būti taikomi ir kombinuoti. Tokios kombinacijos dar vadinamos hibridiniais metodais. Hibridinių informacinių technologijų tarptautiniame žurnalo straipsnyje [44] gauti tokie rezultatai: taikant diskrečiąją bangelių transformaciją, atpažinimo rezultatai prastesni nei 80%, taikant vien HTK pagrindu veikiančią sistemą – apie 90%, o tuo tarpu taikant hibridinę sistemą gaunamas 100% tikslumas. Straipsnyje gauta išvada, kad įvairių kombinuotų, hibridinių metodų rezultatai geresni nei pavienių metodų.

Rezultatai pagerinti 2016 metais. Indijos mokslininkai straipsnyje [4] aprašė mažų matmenų ir mažai galios reikalaujantį biometrinį įrenginį, skirtą kalbėtojo atpažinimui. Produkto veikimo principas pagrįstas įgyvendinant HTK kalbėtojo atpažinimo sistemą. MFCC taikytas kaip kalbėtojų identifikavimo požymis. Kiekvienas kalbėtojas sumodeliuotas kaip vienas paslėptasis Markovo modelis. Patikrinimui taikytas Viterbi dekoderis. Rezultatas – įterptinė sistema pritaikyta kalbos biometrinei sistemai ir sėkmingai įgyvendinta ribotam kalbėtojų skaičiui. Produkto tikslumas – bemaž 100% [4].

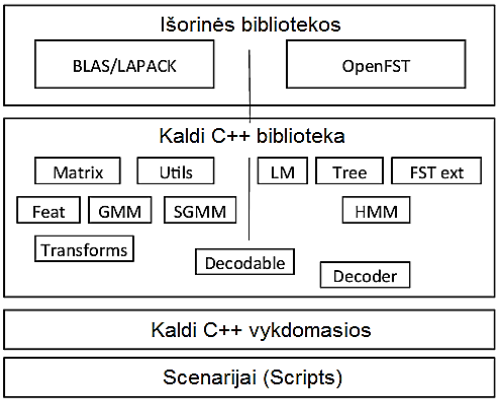
Taigi, HTK sistema – viena seniausių kalbos bei kalbėtojo atpažinimui skirtų sistemų, kurios rezultatai gerinami išleidžiant atnaujintas versijas. Sistemos atnaujinime HTK *Version 3.5 beta* įterptas neuronų tinklų palaikymas, o tai praplėtė sistemos panaudojimo galimybes. Laikui bėgant tobulėjo ne tik ši sistema, bet ir buvo kuriamos analogiškos, tobulesnės nei pradinė HTK sistemos versijos.

1.5.2. Kaldi

Kaldi – C++ kalba parašytas fonemų atpažinimo programinis paketas. Paketo tikslas – modernus ir lankstus visiems prieinamas bei suprantamas kodas. Sistemą sukūrė D. Povey 2011 m. Džono Hopkinso (Johns Hopkins) universitete. Panaudojimo sritis – akustinių modelių mokymas bei kalbos

atpažinimas. Pagrindinis Kaldi skirtumas nuo kitų sistemų – kalbos signalų akustinių modelių sudarymui naudojami neuroniniai tinklai.

Fonemų atpažinimo aplinką sudaro bibliotekos ir mokymo scenarijai. Scenarijuose esančios komandinių eilučių programos praplečia sistemoje esančių bibliotekų taikymo galimybes. Kaldi sistemoje naudojama išorinė biblioteka OpenFST bei tiesinės algebros bibliotekos: BLAS, LAPACK. Kaldi veikimo mechanizmas vaizduojamas žemiau pateiktame paveiksle (žr. 1.11 pav.) [26].



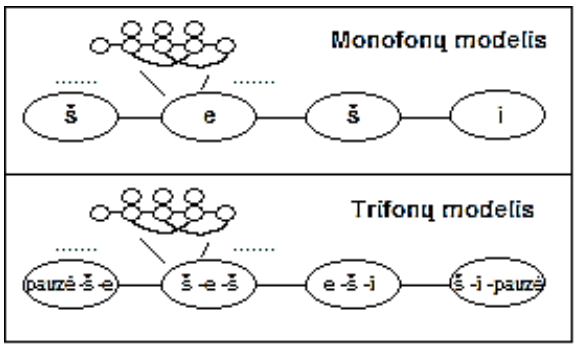
1.11 pav. Kaldi sistemos architektūra

Kaldi taikomi vykdomieji failai, įvedantys įėjimo duomenis iš failų ir įprastai patalpinantys duomenis atgal į failus. Kaip alternatyva vieni sistemos išėjimo duomenys gali būti tiekiami kitai komandai.

Kiekvienai kalbos atpažinimo užduočiai aplinka siūlo alternatyvas: leidžia skaičiuoti požymius [16]: MFCC (apply-mfcc, compute-mfcc-feats), tiesines prognozes (compute-plp-feats) ir t. t.; suteikia galimybę atlikti požymių transformacijas: kepstro vidurkio ir dispersijos normavimas (apply-cmvn) [25], kepstro vidurkio ir dispersijos normuotų statistikų apskaičiavimas (compute-cmvn-stats) [25], tiesinių diskriminanto analizės statistikų kaupimas (acc-lda) [10], požymių transformacija į didesnės dimensijos erdvę (fmpe-apply-transform) [25] ir t. t.

Be to sistemoje pateikiami standartizuoti scenarijai, suteikiantys galimybę pridėti naujų funkcijų. Scenarijai sukaupti Kaldi C++ bibliotekose *utils* arba *steps* ir naudojami tinklo mokymui.

Toliau apibrėžiamos monofonų ir trifonų sąvokos. Žodžio tarimas – tai atskirų vienetų simbolių seka sudaranti žodį. Šie simboliai vadinami fonemomis arba kalbos garsais. Monofoną sudaro vienas kalbos garsas. Trifonai – trijų fonemų junginiai, atsižvelgiantys į greta esančių fonemų kontekstą. Kaldi sistemos veikimo principas pagrįstas trifonų sužymėjimu, jų modelių mokymu bei atpažinimu. Monofono ir trifono modelių pavyzdžiai pateikti 1.12 paveiksle.



1.12 pav. Monofono ir trifono paslėptieji Markovo modeliai lietuvių kalbos žodžiui „šeši“, čia „pauzė“ reiškią tylą kalbėjimo pradžioje ir pabaigoje

Taikant monofonų ir trifonų modelius, atpažinimo tikslumas padidėja, tačiau yra ir trūkumų. Sakykime, kalbą sudaro 20 fonemų, tada skirtingų trifonų kombinacijų būtų $20^3 = 8000$, o, realizuojant atpažinimą realiu laiku, galimų kombinacijų skaičius lemia spartą ir tikslumą [40]. Plačiau apie monofonų ir trifonų modelių sudarymą Kaldi sistemoje analizuojama [37] šaltinyje.

Nemažai eksperimentų atlikta taikant Kaldi sistemą. Straipsnyje [7] apibūdinta, kokių veiksmų imtasi siekiant identifikuoti kalbėtoją bei kalbą. Dėl to straipsnio autoriai sukūrė dirbtinį neuronų tinklą su 4199 būsenos klasteriais, sugeneruotais Kaldi ir Switchboard 1 „tri4a“ duomenų baze. Tinklas sudarytas iš 13 į Gauso klasterius suskirstytų tiesinės prognozės koeficientų, jų pirmos bei antros eilės išvestinių (39 požymiai), 21 lango bei 819 įėjimo požymių. Gauso klasterių modelio kalbos aktyvumo detektoriaus segmentacija taikoma požymiams.

Minėtąjį dirbtinį neuronų tinklą sudaro 7 paslėptieji sluoksniai, kurių kiekviename išsidėstę po 1024 neuronų, išskyrus šeštąjį „butelio kaklelio“ sluoksnį, kuriame 64 neuronai. Visuose paslėptuosiuose sluoksniuose taikoma sigmoido aktyvavimo funkcija, išskyrus šeštąjį sluoksnį, kuriame taikoma tiesinė funkcija.

Eksperimento metu nustatyta, kad geriausi rezultatai grąžinami taikant GMM drauge su „butelio kaklelio“ metodu, o prasčiausi – taikant MFCC drauge su GMM (žr. 1.4 lentelė) [7].

1.4 lentelė. Gauti kalbėtojo atpažinimo rezultatai

Kalbos įrašai	**Metodai**	**Vėlesni**	**EER, %**	**DCF·1000**
Vidiniai	MFCC	Gauso klasterių metodas	2,71	0,404
	MFCC	Neuronų tinklai	2,27	0,336
	Butelio kaklelis	Gauso klasterių metodas	**2,00**	**0,269**
	Butelio kaklelis	Neuronų tinklai	2,79	0,388
Išoriniai	MFCC	Gauso klasterių metodas	6,18	0,642
	MFCC	Neuronų tinklai	3,27	0,427
	Butelio kaklelis	Gauso klasterių metodas	**2,79**	**0,342**
	Butelio kaklelis	Neuronų tinklai	3,97	0,454

Čia DAC –atpažinimo sistema išmokyta ir įvertinta Domain Adaptation Challenge duomenimis.

Straipsnyje [35] aprašytas kalbėtojo atpažinimo tyrimas, taikant vėlinimo neuronų tinklą (TDNN). Kalbėtojo atpažinimui taikytas vienas iš Kaldi sistemos nurodytų būdų, skirtų didelio masto kalbos atpažinimui. Tyrimo metu lyginami gauti rezultatai, ištyrus nuo lyties priklausomus ir nepriklausomus atvejus, taikant Gauso klasterių metodą, prižiūrimą Gauso klasterių metodą (sup-GMM) ir TDNN sistemą. Tyrimo metu gauti rezultatai vaizduojami žemiau pateiktoje lentelėje (žr. 1.5 lentelė).

1.5 lentelė. Tiriant kalbėtojo atpažinimą gauti rezultatai

Modelis	**Sistema**	**EER, %**	**DCF·0.001**	**DCF·0.01**
Nuo lyties priklausomas	Sup-GMM- 5297	1,94	0,388	0,213
	TDNN-5297	**1,20**	**0,210**	**0,123**
	GMM-2048	2,49	0,496	0,288
	GMM-4096	2,56	0,468	0,287
	GMM-5297	2,42	0,484	0,290
Nuo lyties nepriklausomas	Sup-GMM- 5297	1,65	0,354	0,193
	TDNN-5297	**1,09**	**0,214**	**0,108**
	GMM-2048	2,16	0,417	0,239
	GMM-4096	1,96	0,414	0,227
	GMM-5297	2,00	0,410	0,241

Nagrinėjant rezultatus nustatyta, kad, tiriant nuo lyties nepriklausomus duomenis gaunamos mažiausios paklaidos, o komponenčių skaičius veikia rezultatus nežymiai. Likusių sistemų tyrimui taikytos 5297 komponentės. Išsiaiškinta, kad geriausi rezultatai grąžinami taikant TDNN.

Straipsnyje [9] atliktas kalbos atpažinimo sistemų palyginimas ir išsiaiškinta, jog Kaldi lenkia kitas atpažinimo sistemas, kadangi per trumpiausią laiką grąžina geriausius rezultatus. Pastebėta, kad Kaldi ir HTK sistemų sparta skiriasi akivaizdžiai, t. y. Kaldi vartotojai gauna pirmuosius skaičiavimo rezultatus, kai tuo tarpu HTK sistemos vartotojai dar rašo mokymui skirtą kodą.

Taigi, Kaldi privalumas – užduočių ruošiniai. Nors HTK sistemos taikymo galimybės plačios, tačiau požymių paruošimo, duomenų paruošimo, mokymo, rezultatų gavimo ir jų įvertinimo kodų apimtys didelės, o tam reikia didelių laiko sąnaudų. Tuo tarpu Kaldi ruošiniai sutaupo daug laiko, nes atliekant tyrimą pakanka įkelti duomenis reikiamoje vietoje. Vienas didžiausių Kaldi trūkumų – dekoderių stoka. Dinaminis HTK sistemos dekoderis HDecode sudaro žodyną, akustinius modelius. Šiuo atžvilgiu laisvai pasirenkami filtrai Kaldi sistemoje nėra pranašesni.

Apibendrinus galima teigti, kad Kaldi – tai modernus HTK sistemos prototipas, su modernesne vartotojo sąsaja. Nėra tokių požymių HTK sistemoje, kurių nebūtų galima įgyvendinti Kaldi sistemoje, o ir analogiškoms komandoms įgyvendinti sutaupoma laiko. Tačiau ir po Kaldi sukūrimo moksliininkai sparčiai tobulino kalbėtojo atpažinimo technologiją.

1.5.3. MSR Identity Toolbox

MSR Identity Toolbox – tai atviro kodo kalbėtojo atpažinimo sistema, kuri gali būti taikoma ne tik kalbėtojo atpažinimui, bet ir kalbų, dialekto ar akcento nustatymui. Microsoft komanda 2013 m. išleido sistemos versiją *MSR Identity Toolkit v1.0*. Ši sistema laisvai prieinama internetu, adresu: http://research.microsoft.com/jump/203749.

Per pastaruosius 40 metų kalbėtojo atpažinimo sistemos keitėsi iš esmės, t. y. kalbėtojo atpažinimo algoritmai buvo tobulinti nuo vektorių kvantavimo sistemų iki pat GMM ar i-vektoriaus metodo (angl. i-vector method). MSR Identity Toolbox patalpinti MATLAB aplinkos kalbėtojo atpažinimo įrankiai. Šiais įrankiais akustiniai modeliai sudaromi tiek tradiciniais GMM, tiek taikant i-vektoriaus kalbėtojo atpažinimo strategiją.

Atviro kodo kalbėtojo atpažinimo sistemos veikimo principą sudaro 2 etapai. Pirmajame etape išskiriami kalbos požymiai. Požymių išskyrimui būtinos požymių apskaičiavimo, normavimo ir signalo kadrų aktyvumo nustatymo operacijos. Sistemoje nėra nei požymių apskaičiavimo, nei signalo kadrų aktyvumo nustatymo įrankių, todėl sistemos įrankiais realizuojamas tik normavimas. Dėl to išleisti MATLAB aplinkos įrankiai *The Auditory Toolbox* ir *VOICEBOX*, kuriais išskiriami kalbos požymiai bei nustatomas signalo kadrų aktyvumas.

Antrasis kalbėtojo atpažinimo sistemos etapas yra pagrindinis. Šiame etape sudarinėjami ir patikrinami akustiniai modeliai. Akustinių modelių mokymui taikomi GMM arba i-vektoriaus metodai.

Sistemos privalumu laikoma tai, kad kalbėtojo atpažinimo simbolių žymėjimas analogiškas literatūriniam ženklinimui – tai suteikia galimybę vartotojui lengviau suprasti algoritmus. Sistema gali būti kviečiama MATLAB aplinkos komandine eilute arba kompiliuojama be tiesioginės sąsajos su šia aplinka. Be to sistemoje MSR Identity Toolbox patalpinti išskirtų požymių pavyzdžiai, kuriais gali būti tikrinami sudarinėjami modeliai, taikant GMM arba i-vektoriaus metodus. Sistemoje deklaruojamos operacijos [33]:

1) požymių normavimas: kepstro vidurkio ir dispersijos normavimas (cmvn), kepstro vidurkio ir dispersijos normavimas slankiojančio lango atžvilgiu (wcmvn), Gauso normavimas slankiojančiame lange (fea_warping);
2) GMM-UBM sudarymas: taikant EM algoritmą (gmm-em), nustatant didžiausias požymių vektoriaus priklausymo komponentei tikimybes (mapAdapt), GMM metodo įverčių skaičiavimas (score_gmm_trials);
3) i-vektoriaus tikimybių tiesinė diskriminanto analizė: GMM imčių pakankamų statistikų nustatymas (compute_bw_stats), mokymas taikant EM metodą (train_tv_space), i-vektoriaus išskyrimas

(extract_ivector), tiesinė diskriminanto analizė (lda), i-vektoriaus ilgio normavimas, centravimas, vektorių pasiskirstymo diapazono išplėtimas, GMM nustatymas taikant EM (gplda-em), tikimybinė tiesinė diskriminanto analizė (score_gplda_trials);

4) EER ir kompromiso kreivės skaičiavimas: EER (EER), DCF (DCF), kompromiso kreivės skaičiavimas (DET).

Plačiau apie i-vektoriaus metodą, diskriminanto analizę ir kitas nurodytas operacijas kalbama straipsnyje [17]. Toliau darbe apibūdinama kalbos signalų akustinių modelių sudarymo metodika ir organizavimas.

2. KALBOS SIGNALŲ AKUSTINIŲ MODELIŲ SUDARYMO METODIKA IR ORGANIZAVIMAS

Siekiant ištirti kalbos signalų akustinius modelius, tinkančius kalbėtojų atpažinimui, pasirinktos kalbos signalų akustinių modelių tyrimo metodikos, sukurti įvairių kalbų akustiniai modeliai ir atliktas jų tyrimas.

Atliekant tyrimą išskirti MFCC požymiai, rasti akustinių modelių komponenčių pasikartojimo dažniai įrašuose, sudaryti kalbos signalų akustiniai modeliai – atliktas modeliavimas taikant GMM ir k-vidurkių klasterizavimo metodus, atliktas sukurtų kalbos signalų akustinių modelių tyrimas. Šiam tikslui pasiekti Windows operacinėje sistemoje, Anacondos distribucijos Spyder (Python 3.5) aplinkoje, parašytos Python kalbos pusprogramės.

Požymių išskyrimui taikyti anglų, prancūzų, rusų, ispanų, italų ir vokiečių kalbų garso įrašai. Įrašai parinkti įvairių triukšmingumo aplinkų, skirtingo amžiaus ir lyčių asmenų. Be to taikytas UBM failas iš internetinės svetainės www.voicebiometry.org/. Išskirti kalbos požymiai taikyti akustinių modelių sudarymui, o sudarytieji akustiniai modeliai tirti statistiškai.

Toliau aptariamas tyrimo realizavimas: pusprogramių veikimo principai, inicializuojamų bibliotekų ir konfigūracijos parametrų paskirtys, deklaruojamų funkcijų logikos, apibūdinami algoritmai.

2.1. MFCC požymių išskyrimas

Balso biometrija remiasi asmens unikaliomis savybėmis, todėl siekiant atskleisti žmogaus vokalinio trakto fiziologijos ir specifinės kalbėsenos požymius, atliktas tyrimas.

Akustinis modelis priklauso nuo pasirinktų požymių. Tyrimo metu pasirinkti MFCC požymiai, todėl nuskaitinėjant garso įrašus kalbos požymių išskyrimui atlikti šie veiksmai: apskaičiuoti MFCC požymiai, prie gautų rezultatų prijungti skirtuminiai požymiai ir išrikiuoti pagal SFeaCut, nustatyti aktyvūs signalo kadrai, atliktas gautų požymių normavimas, nustatyti GMM didžiausi tikėtinumai, rasti komponenčių pasikartojimo dažniai. Gauti duomenys taikyti komponenčių pasikartojimo dažnių įrašuose nustatymui. Visa tai atlikta 2.1 paveikslo pseodokodu.

Tyrimo etapai: duomenų inicializavimo, funkcijų inicializavimo, požymių išskyrimo, modelio komponenčių pasikartojimo dažnių įrašuose nustatymo. Etapai įgyvendinti taikant požymių išskyrimo *pozymiu_isskyrimas.py* (žr. 1 priede) ir statistikų apskaičiavimo *statistiku_skaiciavimas.py* (žr. 2 priede) pusprogrames.

```
Importuojamos bibliotekos
FILES = sudaromas pasirinkto aplanko garso įrašų failų sąrašas
Inicializuojami konfigūracijos parametrai
Ubm_file = vykdomas UBM failo nuskaitymas
GMM = kviečiama GMM modelio inicializavimo funkcija
FOR garso įrašo failas iš sąrašo FILES
        signalas, dažnis = vykdomas įrašo nuskaitymas
        IF (dažnis nelygu 8000) THEN
                Pakeičiamas signalo atskaitų dažnis į 8000
        END IF
        MFCC požymiai = kviečiama MFCC požymių apskaičiavimo funkcija
        MFCC požymiai = prijungiami skirtuminiai požymiai
        MFCC požymiai = išrikiuojame MFCC požymius pagal SFeaCut
        Aktyvaus signalo = kviečiama signalo kadrų aktyvumo nustatymo funkcija
        MFCC požymiai = atliekamas MFCC požymių normavimas
        GMM tikėtinumo reikšmės = randamos GMM komponenčių tikėtinumo reikšmės
        indeksai = surandami indeksai GMM komponenčių su maksimaliomis tikėtinumo reikšmėmis
        LLK = randamos indeksus atitinkančios GMM tikėtinumo reikšmės
        Balso sričių komponenčių indeksai = surašomi balso sričių signalo kadrų GMM komponenčių indeksai
        Fono sričių komponenčių indeksai = surašomi fono sričių signalo kadrų GMM komponenčių indeksai
        Balso sričių komponenčių dažnumas = išskiriama ir suskaičiuojama kiek kartų pasikartoja balso sričių
        komponentės
        Fono sričių komponenčių dažnumas = išskiriama ir suskaičiuojama kiek kartų pasikartoja fono sričių
        komponentės
        Rezultatų išsaugojimas
END FOR
Balso sričių komponenčių statistikos skaičiavimas
Fono sričių komponenčių statistikos skaičiavimas
Randamos balso ir fono sričių komponenčių pasikartojančios reikšmės
Randamos balso ir fono sričių komponenčių procentinės išraiškos
Randami procentinių išraiškų kvartiliai
Rezultatų išsaugojimas
```

2.1 pav. Komponenčių statistikos tyrimo pseudokodas

Toliau pateikiama komponenčių statistikos tyrimo pseudokodą realizuojančių etapų analizė, t. y. nurodoma bibliotekų ir parametrų paskirtis, pateikiami deklaruojamų funkcijų aprašai ir veikimo principas, detalizuojama pseudokodo logika.

2.1.1. Duomenų inicializavimas

Duomenų inicializavimo etape inicializuojami konfigūracijos ir loginiai parametrai, importuojamos bibliotekos ir pagalbinės pusprogramės, sudaromas pasirinkto aplanko garso įrašų failų sąrašas *FILES*, nuskaitomas UBM failo vardas, nuskaitomi statistiniai duomenys: komponenčių vidurkiai ir standartiniai nuokrypiai.

Importuotos bibliotekos: *os* – įvairių operacinių sąsajų, *numpy* – tiksliųjų mokslų operacijų, *scipy.io.wavfile* – wav tipo failų nuskaitymo/įrašymo, *scipy.signal* – signalų apdorojimo, *glob* – globalių kintamųjų sąrašo sudarymo, *Counter* – unikalių kintamųjų pasikartojimų skaičiaus nustatymo.

Importuotos pagalbinės pusprogramės: *MFCC_pozymiai.py* – MFCC požymių apskaičiavimo, *gmm.py* – GMM nustatymo ir *signalo_kadru_aktyvumas.py* – signalo kadrų aktyvumo nustatymo (žr. 3 priede). Pusprogramėse deklaruojamos funkcijos kviečiamos požymių išskyrimo etape.

Statistinių duomenų taikymui sudaromi indeksų sąrašai: *columns_idx* – nuo 0 iki 59, neįterpiant 0-nio ir 3-iojo indeksų, *columns_rest_idx* – nuo 2 iki 57. Indeksų sąrašai taikomi komponenčių radimo etape normuojant požymius, kur 0-inis ir 3-iasis požymiai atskiriami ir normuojami pagal failo duomenis, o likusieji požymiai normuojami pagal statistinius duomenis. 0-inis ir 3-iasis požymiai normuojami atskirai, nes 0-inis požymis priklauso nuo kalbėjimo garsumo arba kaip arti esama mikrofono, o 3-iasis požymis priklauso nuo mikrofono. Nustatomos indeksų sąrašą *columns_idx* atitinkančios komponenčių vidurkių bei standartinių nuokrypių reikšmės. Toliau apžvelgiami konfigūracijos parametrai, loginiai parametrai bei nurodoma jų paskirtis.

Teorinėje dalyje, apžvelgiant atviro kodo fonemų atpažinimo sistemas, išsiaiškinta, kad MFCC parametrų skaičiavimui plačiai naudojama HTK sistema. Darbe skaičiuojamus MFCC požymius charakterizuosime HTK sistemos konfigūracijos parametrais (žr. 2.1 lentelė).

2.1 lentelė. Požymių išskyrimo pusprogramės konfigūracijos parametrai

Konfigūracijos parametras	Reikšmė	Paskirtis
SOURCERATE	1250	Atskaitų periodas (HTK laiko vienetais)
TARGETRATE	100000	Laiko intervalai (HTK laiko vienetais)
LOFREQ	120	Žemųjų dažnių riba (Hz)
HIFREQ	3800	Aukštųjų dažnių riba (Hz)
WINDOWSIZE	250000	Lango ilgis (HTK laiko vienetais)
PREEMCOEF	0,97	Pradinio filtravimo koeficientas
NUMCHANS	24	Filtrų rinkinio filtrų skaičius (vnt.)
CEPLIFTER	22	Kepstro filtrų skaičius (vnt.)
NUMCEPS	19	Kepstro koeficientų skaičius
deltawindow, accwindow	2	Skirtuminių požymių skaičiavimo langas
cmvn_lc	150	Kadrų skaičius iš kairės nuo slankiojančio lango esamos padėties (vnt.)
cmvn_rc	150	Kadrų skaičius iš dešinės nuo slankiojančio lango esamos padėties (vnt.)
fs	8000	Imties dažnis (Hz)
window	200	Kadro lango ilgis atskaitomis
noverlap	120	Persidengimų tarp gretimų kadrų ilgis atskaitomis
ESCALE	0,1	Logaritminės energijos mastelis
SILFLOOR	50	Energijos apatinė slenkstinė riba (dB)

Vadinasi, turime garso signalus diskretizuotus 8 kHz dažniu. Atskaitų periodas – 125 µs. HTK sistemoje laikas matuojamas 100 ns intervalais. Todėl atskaitų periodas 125 µs yra lygus 1250 HTK laiko intervalų, o lango ilgį sudaro 250000 HTK laiko intervalų.

Be to inicializuojami konfigūracijos parametrai: žemųjų dažnių riba – 120 Hz, aukštųjų dažnių riba – 3800 Hz, pradinio filtravimo koeficientas – 0,97, filtrų rinkinio filtrų skaičius – 24, kepstro filtrų skaičius – 22, kepstro koeficientų skaičius – 19, skirtuminių požymių skaičiavimo langas – 2, kadrų skaičius tiek iš kairės, tiek iš dešinės pusės nuo slankiojančio lango esamos padėties – po 150 atskaitų, imties dažnis – 8 kHz, kadro lango ilgis – 200 atskaitų, persidengimų tarp gretimų kadrų ilgis – 120 atskaitų, logaritminės energijos mastelis – 0,1, energijos apatinė slenkstinė riba – 50.

Loginiais parametrais (žr. 2.2 lentelė) vykdomos komandos: vidurkio atėmimo iš signalo, Hammingo lango naudojimo, energijos požymio grąžinimo, energijos normavimo.

2.2 lentelė. Loginiai parametrai

Loginiai parametrai	Aktyvacija	Paskirtis
ZMEANSOURCE	TRUE	Vidurkio atėmimas iš signalo
USEHAMMING	TRUE	Hammingo lango naudojimas
RAWENERGY	TRUE	Energijos požymio grąžinimas
ENORMALISE	TRUE	Energijos normavimas

Loginiai parametrai taikomi kviečiant požymių apskaičiavimo pusprogramės MFCC požymių apskaičiavimo funkciją.

2.1.2. Funkcijų inicializavimas

Pusprogramėse deklaruojamos funkcijos, kurios kviečiamos vykdant požymių išskyrimo etapą. Funkcijos apibrėžiamos pusprogramėse: MFCC požymių apskaičiavimo, GMM požymių apskaičiavimo ir apdorojimo, signalo kadrų aktyvumo nustatymo ir požymių išskyrimo.

MFCC požymių apskaičiavimo pusprogramėje deklaruotos funkcijos pateikiamos 2.3 lentelėje.

2.3 lentelė. MFCC požymių apskaičiavimo pusprogramės funkcijos

Funkcija	Paskirtis
mel_fbank_mx	Mel filtrų rinkinio apskaičiavimas
mel	Mel transformacija
mel_inv	Atvirkštinė Mel transformacija
mfcc_htk	MFCC požymių apskaičiavimas
framing	Konvertuoja kalbos signalą į kadrus
dct_basis	DCT
preemphasis	Kalbos signalo filtravimas
add_deriv	MFCC požymių papildymas skirtuminiais požymiais
cmvn_floating	MFCC požymių normavimas

Mel filtrų rinkinio apskaičiavimo funkcija grąžina Mel filtrų rinkinį. Parametrai: kadro langa ilgis atskaitomis, imties dažnis, filtrų rinkinio filtrų skaičius, žemųjų dažnių riba, aukštųjų dažnių riba, Mel transformacija, atvirkštinė Mel transformacija. Algoritmas: konfigūruojama aukštųjų dažnių riba, vykdoma FFT, atliekamos Mel ir atvirkštinė Mel transformacijos, kaupiamos Mel filtrų rinkinio reikšmės.

Kadangi funkcijos įgyvendinimui reikalingos Mel ir atvirkštinės Mel transformacijos – deklaruojamos Mel ir atvirkštinės Mel transformacijos funkcijos. Mel transformacijos funkcija kalbos signalą $s(t)$ transformuoja į MFCC požymius:

$$fea = 1127 \cdot \log(1 + \frac{s(t)}{1127}), \tag{2.1}$$

o atvirkštinės Mel transformacijos funkcija MFCC požymius transformuoja į kalbos signalą:

$$s(t) = (e^{\frac{fea}{1127}} - 1) \cdot 700. \tag{2.2}$$

Čia fea – MFCC požymiai, $s(t)$ – kalbos signalas.

MFCC požymių apskaičiavimo funkcija grąžina MFCC požymius. Parametrai: kalbos signalas $s(t)$, langa ilgis atskaitomis, persidengimų tarp gretimų kadrų ilgis atskaitomis, Mel filtrų rinkinys, kepstro koeficientų skaičius, energijos požymio grąžinimas, pradinio filtravimo koeficientas, kepstro filtrų skaičius, vidurkio atėmimas iš signalo, energijos normavimas, logaritminės energijos mastelis, energijos apatinė slenkstinė riba, Hammingo lango naudojimas. Algoritmas: nustatomi ir normuojami DCT koeficientai, kalbos signalas $s(t)$ konvertuojamas į kadrus, signalo kadrai padauginami iš Hammingo lango funkcijos, skaičiuojami kiekvieno kadro DFT, signalo spektras dauginamas iš Mel skalės filtrų rinkinių, gauti rezultatai logaritmuojami ir taikoma DCT.

Funkcijos realizavimui apibrėžiamos kalbos signalo konvertavimo į kadrus, DCT ir signalo filtravimo funkcijos.

Kalbos signalo konvertavimo į kadrus funkcija grąžina kalbos signalo kadrus. Taikomi konfigūracijos parametrai: kalbos signalas $s(t)$, langa ilgis atskaitomis, persidengimų tarp gretimų kadrų ilgis atskaitomis. Funkcija pagal kadrų skaičių, kadrų ilgį ir persidengimų ilgį signalą konvertuoja į kadrus. Kadrų skaičius nustatomas lygybe

$$\text{frames} = \frac{\sum_{t=0}^{T-1} s(t) - \text{window}}{\text{window} - \text{noverlap}}. \tag{2.3}$$

Čia T – signalo $s(t)$ ilgis, frames – kadrų skaičius, window – kadrų ilgis, noverlap – persidengimų tarp gretimų kadrų ilgis.

DCT funkcija grąžina DCT koeficientus, kurių amplitudės atidedamos pagal dažnį. Konfigūracijos parametrai: kepstro koeficientų skaičius, Mel filtrų rinkinio ilgis. Funkcija vykdo DCT.

Signalo filtravimo funkcijos paskirtis – pašalinių triukšmų ir iškraipymų pašalinimas signale. Parametrai: kalbos signalas $s(t)$, pradinio filtravimo koeficientas. Sakykime, turime tokį kalbos signalą $s(t)$ ir kalbos signalo filtravimo vektorių $c(t)$:

$$s(t) = (s(0), s(1), s(2), s(3), \dots, s(T-2), s(T-1)) \tag{2.4}$$

$$c(t) = (s(0), s(0), s(1), s(2), s(3), \dots, s(T-3), s(T-2)). \tag{2.5}$$

Tada, filtruotas signalas $s_{\text{filtr}}(t)$ nustatomas tokiu būdu:

$$s_{\text{filtr}}(t) = s(t) - c(t) \cdot \text{PREEMCOEF}. \tag{2.6}$$

MFCC požymių papildymo skirtuminiais požymiais funkcija grąžina papildytą požymių sąrašą. Funkcijos parametrai: MFCC požymiai, skirtuminių požymių skaičiavimo langas. MFCC papildomi skirtuminiais požymiais tokiu būdu: pirma – apskaičiuojami MFCC skirtuminiai požymiai, t. y. randamos požymių išvestinės, antra – MFCC požymių matrica papildoma skirtuminiais požymiais.

MFCC požymių normavimo funkcija grąžina normuotus požymius. Parametrai: MFCC požymiai, kadrų kiekis iš kairės ir iš dešinės pusės nuo slankiojančio lango esamos padėties. Normavimo algoritmas pateikiamas 1.3.1. skyrelyje.

Toliau aptariama GMM požymių apskaičiavimo ir apdorojimo pusprogramė. Pusprogramėje apibrėžiamos pagalbinės matricų algebros funkcijos pateikiamos 2.4 lentelėje.

2.4 lentelė. GMM požymių apskaičiavimo ir apdorojimo pusprogramės pagalbinės funkcijos

Funkcija	**Paskirtis**
uppertri_indices	Grąžina viršutinės trikampės matricos eilučių bei stulpelių indeksus
uppertri_to_sym	Dvimatę viršutinę trikampę matricą transformuoja į trimatę simetrinę matricą
uppertri1d_from_sym	Trimatę simetrinę matricą transformuoja į vienmatę matricą
uppertri1d_to_sym	Vienmatę matricą transformuoja į trimatę simetrinę matricą
inv_posdef_and_logdet	Grąžina logaritmo determinantą ir atvirkštinę matricą

Visos šios funkcijos taikomos kviečiant GMM modelio inicializavimo funkciją. Plačiau apie matricų algebrą suvesta knygoje [42].

Be to pusprogramėje deklaruojamos pagrindinės funkcijos. Jų paskirtis nurodoma 2.5 lentelėje.

2.5 lentelė. GMM požymių apskaičiavimo ir apdorojimo pusprogramės pagrindinės funkcijos

Funkcija	**Paskirtis**
gmm_eval_prep	GMM modelio inicializavimas
gmm_llhs	GMM tikėtinumų įvertinių nustatymas
gmm_eval	GMM modelio parametrų išskyrimas
gmm_update	GMM parametrų atnaujinimas
logsumexp	Eksponenčių sumų logaritmų apskaičiavimas

GMM modelio inicializavimo funkcija grąžina GMM modelį GMM, t. y. modeliui priskiriami parametrai. Įvesties parametrai: GMM svorių koeficientai c, vidurkių vektorius μ ir kovariacijų matrica Σ. Funkcijos veikimo principas: nuskaitomi ir modeliui priskiriami įvesties parametrai, taikant matricų algebrą. Funkcija kviečiama signalo kadrų aktyvumo nustatymo ir požymių išskyrimo pusprogramėse.

GMM tikėtinumų įvertinių nustatymo funkcija grąžina GMM tikėtinumų įvertinius $\widehat{gamma}$. Parametrai: signalo energija E ir GMM modelis GMM. GMM tikėtinumų įvertiniai

$$\widehat{gamma} = -\frac{1}{2} \cdot GMM^2 \cdot \Sigma^T + E \cdot \mu^T + c. \tag{2.7}$$

GMM modelio parametrų išskyrimo funkcijos įvesties parametrai: signalo energija E, GMM modelis GMM. Vykdant funkciją nustatomi ir grąžinami GMM logaritminiai tikėtinumai, GMM svorių koeficientai, vidurkių vektorius ir kovariacijų matrica:

$$\log(gamma) = logsumexp(\widehat{gamma}), \tag{2.8}$$

$$c = \sum_{m=0}^{M-1} e^{\widehat{gamma}_{jm}{}^T - \log L(\theta)}, \tag{2.9}$$

$$\mu = E \cdot e^{\widehat{gamma}^T - \log L(\theta)}, \tag{2.10}$$

$$\Sigma = GMM^2 \cdot e^{\widehat{gamma}^T - \log L(\theta)}, \tag{2.11}$$

čia $\widehat{gamma}_{jm}$ – j-tosios eilutės m-tojo stulpelio GMM tikėtinumo įvertinys, $\log(gamma)$ – GMM logaritminiai tikėtinumai, c – svorių koeficientai, μ – vidurkių vektorius ir Σ – kovariacijų matrica.

Eksponenčių sumų logaritmų apskaičiavimo funkcija nustato eksponenčių sumų logaritmus. Sakykime, funkcijos įvesties parametras x. Tada grąžinama reikšmė nustatoma tokiu būdu:

$$\text{logsumexp}(x) = x_{max} + \log\left(\sum_{j=0}^{J-1} \text{e}^{x_{jm} - x_{j_{max}}}\right). \tag{2.12}$$

Čia $x_{j_{max}}$ – įvesties parametro j-tosiso eilutės maksimali reikšmė, x_{max} – įvesties parametro eilučių maksimalios reikšmės, x_{jm} – j-tosios eilutės m-tojo stulpelio įvesties parametro matricos elementas, $0 \le j \le J-1$, $0 \le m \le M-1$, J – eilučių skaičius, M – stulpelių skaičius.

GMM parametrų atnaujinimo funkcija grąžina atnaujintas svorių koeficientų c_{update}, vidurkių vektoriaus μ_{update} ir kovariacijų matricos Σ_{update} reikšmes. Konfigūracijos parametrai: svorių koeficientai c, vidurkių vektorius μ ir kovariacijų matrica Σ. GMM parametrai atnaujinami išraiškomis:

$$c_{update} = \frac{c}{\sum_{m=0}^{M-1} c_{jm}}, \quad \mu_{update} = \frac{\mu}{c}, \quad \Sigma_{update} = \frac{\Sigma}{c - \mu_{update} \times \mu_{update}}. \tag{2.13}$$

Čia c_{jm} – komponentės j būsenoje m svorio koeficientas, $0 \le j \le J-1$, $0 \le m \le M-1$, J – komponenčių skaičius, M – būsenų skaičius.

GMM tikėtinumo įvertinių nustatymo, GMM modelio parametrų išskyrimo ir GMM parametrų atnaujinimo funkcijos taikomos signalo kadrų aktyvumo nustatymo pusprogramėje.

Signalo kadrų aktyvumo nustatymo pusprogramėje deklaruojamos funkcijos pateikiamos 2.6 lentelėje.

2.6 lentelė. Signalo kadrų aktyvumo nustatymo pusprogramės funkcijos

Funkcija	Paskirtis
framing	Konvertuoja kalbos signalą į kadrus
compute_vad	Signalo kadrų aktyvumo nustatymas

Signalo kadrų aktyvumo nustatymo funkcija grąžina signalo aktyvius kadrus vad. Įvesties parametrai: kalbos signalas $s(t)$, kadro lango ilgis atskaitomis, persidengimų tarp gretimų kadrų ilgis atskaitomis. Algoritmas:

1) nustatoma signalo energija E. Dėl to signalo galia $s^2(t)$ konvertuojama į kadrus – kviečiama signalo konvertavimo į kadrus funkcija, apskaičiuojama signalo galios kadrų suma;
2) vykdomas energijos E normavimas;
3) inicializuojamas GMM modelis – kviečiama GMM modelio inicializavimo funkcija;
4) nustatomi modelio tikėtinumai, svoriai, vidurkiai ir kovariacijos – kviečiama GMM modelio parametrų išskyrimo funkcija;
5) pagal rastas reikšmes atnaujinami parametrai – kviečiama GMM modelio atnaujinimo funkcija;
6) modeliui priskiriami atnaujinti parametrai – kviečiama GMM modelio inicializavimo funkcija;
7) nustatomi GMM tikėtinumų įvertiniai $\widehat{gamma}$ – kviečiama GMM tikėtinumų įvertinių nustatymo funkcija;
8) randami GMM logaritminiai tikėtinumai $\log(gamma)$ – GMM tikėtinumų įvertiniams $\widehat{gamma}$ kviečiama eksponenčių sumų logaritmo apskaičiavimo funkcija;
9) nustatomi GMM logaritminių tikėtinumų įvertiniai

$$\log(\widehat{gamma}) = e^{\widehat{gamma} - \log(gamma)}. \quad (2.14)$$

10) nustatomi signalo aktyvūs kadrai – loginio parametro vad reikšmės teisingos, kai

$$\log(\widehat{gamma}) < 0{,}3. \quad (2.15)$$

Signalo konvertavimo į kadrus funkcijos veikimo principas analogiškas MFCC požymių apskaičiavimo pusprogramės konvertavimo į kadrus funkcijai. Skiriasi tik parametrų reikšmės: čia kadro lango ilgis – 160 atskaitų, o persidengimų tarp gretimų kadrų ilgis – 80 atskaitų.

MFCC požymių išskyrimo pusprogramėje deklaruojamos funkcijos pateikiamos 2.7 lentelėje.

2.7 lentelė. Požymių išskyrimo pusprogramės funkcijos

Funkcija	Paskirtis
load_ubm	UBM failo nuskaitymas
compute_vad	Signalo kadrų aktyvumo nustatymas

UBM failo nuskaitymo funkcija kviečiama nuskaitinėjant UBM failą. Įvesties parametras: failo vardas. Algoritmas: pagal nurodytą vardą nuskaitomas tekstinis failas, o pagal nuskaitytus duomenis grąžinami GMM parametrai: svorių koeficientai c, vidurkių vektorius μ ir kovariacijų matrica Σ.

Signalo kadrų aktyvumo nustatymo funkcija taikoma signalo aktyvių kadrų nustatymui. Funkcijos parametrai: kalbos signalas $s(t)$, kadro lango ilgis – 160 atskaitų, persidengimų tarp gretimų kadrų ilgis – 80 atskaitų. Nustatinėjant aktyvius signalo kadrus kviečiama signalo kadrų aktyvumo nustatymo pusprogramės signalo kadrų aktyvumo nustatymo funkcija, apskaičiuojama rastų signalo aktyvių kadrų suma. Funkcija grąžina signalo aktyvius kadrus vad ir jų sumą.

Toliau apibūdinamas kalbos požymių išskyrimo etapas.

2.1.3. Požymių išskyrimas

Požymių išskyrimo etapas – vienas svarbiausių kalbėtojų atpažinimo procese. Nuo požymių išskyrimo tiesiogiai priklauso identifikavimo tikslumas.

Siekiant išskirti garso įrašų požymius ir nustatyti modelio komponentes, vykdomas parengiamasis duomenų apdorojimas: kviečiama Mel filtrų rinkinio apskaičiavimo funkcija – grąžinami Mel filtrų rinkiniai, vykdomas kalbos signalų akustinio modelio nuskaitymas. Po to GMM modeliui priskiriami nuskaityto akustinio modelio parametrai – kviečiama GMM modelio inicializavimo funkcija, vykdomas modelio statistikų normavimas.

Apdorojus parengiamuosius duomenis nuskaitinėjami garso įrašų failai iš sąrašo *FILES*, nuskaitomas įrašo dažnis ir garso signalas, vykdomas požymių išskyrimas. Etapas kartojamos tol, kol išskiriami kalbos požymiai visiems garso įrašų failams.

Nuskaitinėjant įrašų duomenis, signalo atskaitų dažnis keičiamas į 8 kHz, todėl vykdoma konfigūracija: grąžinamas pirmasis įrašo signalo stulpelis, jei signalo dydis ne mažesnis nei 1 bei pakeičiamas signalo atskaitų dažnis į 8 kHz, jei nėra lygus 8 kHz. Keičiant signalo dažnį randama signalo trukmė ir imties dydis.

Toliau nustatomi: dažnis, signalo ilgis, MFCC požymiai – kviečiama MFCC požymių apskaičiavimo funkcija, gauti požymiai papildomi pirmos ir antros eilės skirtuminiais požymiais – šiam

tikslui pasiekti kviečiama MFCC požymių papildymo skirtuminiais požymiais funkcija. Pridėjus skirtuminius požymius rezultatai išrikiuojami pagal SFeaCut, nustatomi aktyvūs signalo kadrai – kviečiama signalo kadrų aktyvumo nustatymo funkcija.

Nustačius aktyvius signalo kadrus, atliekamas MFCC požymių normavimas: 0-inis ir 3-iasis požymiai atskiriami ir normuojami pagal failo duomenis – kviečiama MFCC požymių normavimo funkcija, likę požymiai normuojami pagal statistinius duomenis – požymių ir komponenčių vidurkių atėminys padalijamas iš komponenčių standartinių nuokrypių. Po normavimo požymiai sustatomi ankstesniąja tvarka.

Taikant normuotus požymius nustatomi: GMM komponenčių tikėtinumai, GMM didžiausių tikėtinumų indeksai, indeksus atitinkančios GMM tikėtinumų reikšmės. GMM tikėtinumų apskaičiavimui vykdomas kvadratinis požymių išplėtimas.

Gauti GMM didžiausių tikėtinumų indeksai taikomi atskiriant balso sričių komponentes nuo fono sričių komponenčių bei balso sričių požymių nustatymui. Dėl to vykdomas algoritmas:

1) sukuriama loginė matrica *mask_power*, kuri teisinga, kai požymių matricos pirmo stulpelio elementai didesni už 30;
2) randami balso sričių indeksai – sukuriama loginė matrica $mask$, kuri teisinga, kai signalo kadras aktyvus ir loginė matrica $mask\_power$ teisingi;
3) nustatomi GMM balso sričių indeksai – pagal loginės matricos *mask* teisingas reikšmes surašomi balso sričių indeksai;
4) nustatomi fono sričių indeksai – randama atvirkštinė matrica $mask^{-1}$;
5) nustatomi fono sričių GMM indeksai – pagal atvirkštinės matricos $mask^{-1}$ teisingas reikšmes surašomi fono sričių indeksai;
6) nustatomi balso sričių požymiai – pagal loginės matricos *mask* teisingas reikšmes surašomi balso sričių indeksai.

Nustačius įrašų balso ir fono sričių komponentes bei balso sričių požymius, vykdomas balso ir fono sričių komponenčių pasikartojimų apskaičiavimo etapas.

Šiame etape apskaičiuojami balso ir fono sričių komponenčių pasikartojimo dažniai – kviečiama unikalių pasikartojimų skaičiavimo bibliotekos *Collections* funkcija *counter*. Ši funkcija nustato, kiek kartų pasikartoja kiekvienas unikalus imties narys. Grąžinamas unikalių modelio komponenčių pasikartojimų skaičius. Gauti rezultatai klasifikuojami ir išsaugomi. Duomenys grupuojami atskiriant moterų įrašų rezultatus nuo vyrų.

2.1.4. Modelio komponenčių pasikartojimo dažnių įrašuose nustatymas

Komponenčių pasikartojimo dažnių įrašuose nustatymo etape randamos moterų ir vyrų įrašų balso ir fono sričių pasirinkto modelio komponenčių statistikos, apskaičiuojamos komponenčių pasikartojančios reikšmės, nustatomos komponenčių užimamų sričių procentinės išraiškos. Etapo logika pateikiama statistikų apskaičiavimo pusprogramėje (žr. 2 priedas).

Inicializuojant pusprogramės įvesties duomenis pagal tiriamą akustinį modelį sudaromi sąrašai: *FILES_voice_f* – moterų balso sričių komponenčių aplanko failų sąrašas, *FILES_voice_m* – vyrų balso sričių komponenčių aplanko failų sąrašas, *FILES_bg_f* – moterų fono sričių komponenčių aplanko failų sąrašas, *FILES_bg_m* – vyrų fono sričių komponenčių aplanko failų sąrašas.

Inicializavus įvesties duomenis, apskaičiuojamos moterų ir vyrų, balso ir fono sričių pasirinkto modelio komponenčių statistikos, randamos komponenčių pasikartojančios reikšmės įrašuose. Gauti rezultatai taikomi komponenčių pasikartojimų dažnių įrašuose apskaičiavimui. Modelio komponenčių procentinės išraiškos randamos taikant matematinę proporciją.

Gauti rezultatai struktūrizuojami – nustatomi komponenčių pasikartojimų dažnių įrašuose kvartiliai: $Q_0 = 0, Q_1 = 0{,}25, Q_2 = 0{,}5, Q_3 = 0{,}75, Q_4 = 1$. Kvartilių rezultatai išrikiuojami medianos (Q_2) atžvilgiu. Rezultatai išsaugomi.

2.2. Akustinio modelio sudarymas taikant GMM

Kalbos signalų akustinių modelių sudarymui taikytas GMM metodas. Metodas realizuojamas pusprograme *GMM_mokymas.py* (žr. 4 priede). Pusprogramės logika pateikiama 2.2 paveiksle.

```
Importuojamos bibliotekos
FILES = vykdomas pasirinkto aplanko požymių nuskaitymas
Inicializuojami konfigūracijos parametrai
FOR modelio didinimo iteracijos
    FOR modelio tikslinimo iteracijos
        FOR balso srities požymiai iš sąrašo FILES
            Vykdomas įrašo požymių nuskaitymas
            Kaupiamos svorių, vidurkių, kovariacijų ir tikėtinumų reikšmės
        END FOR
        GMM = kviečiama maksimizavimo funkcija
    END FOR
    Rezultatų išsaugojimas
    GMM = kviečiama komponenčių skaičiaus padidinimo funkcija
END FOR
```

2.2 pav. Akustinio modelio sudarymo taikant GMM pseudokodas

Taigi, GMM modelio sudarymo etapai: duomenų inicializavimo, funkcijų inicializavimo, GMM mokymo. Kalbos signalų akustinis modelis sudaromas įvykdžius visas modelio tikslinimo ir didinimo iteracijas.

2.2.1. Duomenų inicializavimas

Duomenų inicializavimo etape atliekamas reikiamų bibliotekų importavimas (*fnmatch*, os, *numpy*), inicializuojami įvesties duomenys ir konfigūracijos parametrai.

Importuojamų bibliotekų paskirtis – elementarių operacijų vykdymo ir duomenų apdorojimo galimybės. Taigi, *fnmatch* ir *os* taikomos vykdant pasirinkto aplanko požymių nuskaitymą, *numpy* taikoma vykdant elementarias operacijas.

Inicializuojant įvesties duomenis iš pasirinkto aplanko sudaromas moterų ir vyrų balso sričių požymių sąrašas *FILES*. Konfigūracijos parametrai pateikiami 2.8 lentelėje.

2.8 lentelė. Kalbos signalų akustinių modelių sudarymo taikant GMM konfigūracijos parametrai

Parametras	Reikšmės	Paskirtis
nFiles	2048	Komponenčių skaičius
nFeatures	60	Požymių skaičius
var_const	0,1	Kovariacijų matricos slenkstinė riba
mixList	1, 2, 4, 8, 16, 32, 64, 128, 256, 512, 1024, 2048	Mišinio komponenčių sąrašas
niter	1, 2, 4, 4, 4, 4, 6, 6, 10, 10, 15, 15	Iteracijų sąrašas

Vadinasi, modelis sudaromas 2048 komponentėms ir 60 požymių. Mišinio komponenčių sąrašas taikomas kviečiant modelio bendrinimo funkciją. Iteracijų sąrašą sudaro 12 narių, todėl vykdomas 12 iteracijų mokymas. GMM kovariacijų matricos filtravimui parenkama slenkstinė riba – 0,1.

Toliau aptariamos pusprogramėje deklaruojamos kalbos signalų akustinių modelių sudarymo funkcijos.

2.2.2. Funkcijų inicializavimas

GMM mokymo pusprogramėje pateikiamos funkcijos, kurios taikomos GMM modelio sudarymui, t. y. parametrų nustatymui ir modelio bendrinimui. GMM parametrų nustatymui vykdomas 1.3.2. skyrelyje aptartas Baum-Welch algoritmas, kurio didžiausio tikėtinumo įvertiniams apskaičiuoti taikomas EM algoritmas. Funkcijų paskirtis nurodoma 2.9 lentelėje.

2.9 lentelė. Kalbos signalų akustinių modelių sudarymo taikant GMM pusprogramės funkcijos

Funkcija	Paskirtis
expectation	Nustato GMM tikėtinumus
maximization	Nustato didžiausius GMM tikėtinumus
postprob	Grąžina požymių vektoriaus tikimybes, GMM logaritminius tikėtinumus
lgmmprob	Požymių logaritminių tikėtinumų nustatymas pagal duotą GMM
logsumexp	Eksponenčių sumų logaritmų apskaičiavimas
apply_var_floors	Vykdo kovariacijų matricos filtravimą
gmm_mixup	GMM modelio bendrinimas

GMM tikėtinumų nustatymo funkcija grąžina GMM logaritminius tikėtinumus $\log(gamma)$, GMM svorių koeficientus c, vidurkių vektorių μ ir kovariacijų matricą Σ. Parametrai: balso sričių požymiai fea, GMM modelis GMM. Algoritmas: apskaičiuojamos tikimybės $post$, kad požymių vektorius priklauso komponentei, nustatomi GMM logaritminiai tikėtinumai $\log(gamma)$ – kviečiama požymių vektoriaus tikimybių ir GMM logaritminių tikėtinumų nustatymo funkcija*;* randami svoriai c, vidurkiai μ ir kovariacijos Σ:

$$c = \sum_{m=0}^{M-1} post_{jm}{}^{T}, \quad \mu = fea \cdot post^{T}, \quad \Sigma = (fea \times fea) \cdot post^{T}. \tag{2.16}$$

Čia $post_{jm}$ – j-osios eilutės m-tojo stulpelio tikimybių matricos elementas, M – požymių vektoriaus tikimybių matricos stulpelių skaičius.

Funkcijos vykdymui reikalinga tikimybių $post$ ir GMM logaritminių tikėtinumų $\log(gamma)$ nustatymo funkcija. Funkcija nustato tikimybes, kad požymių vektorius priklauso komponentei. Parametrai: požymiai fea, GMM vidurkiai μ, kovariacijos Σ ir svoriai c. Funkcijos logika: nustatomi požymių logaritminiai tikėtinumai $\log(fea_{prob})$ – kviečiama požymių logaritminių tikėtinumų nustatymo pagal duotą GMM modelį funkcija, nustatomi ir grąžinami: GMM logaritminiai tikėtinumai $\log(gamma)$ – gautiems požymių logaritminiams tikėtinumams $\log(fea_{prob})$ kviečiama eksponenčių sumų logaritmų apskaičiavimo funkcija (2.12). Tada požymių vektoriaus tikimybės

$$post = e^{\log(fea_{prob}) \; - \log L(\theta)}. \tag{2.17}$$

Požymių logaritminių tikėtinumų nustatymo pagal duotą GMM funkcija grąžina požymių logaritminius tikėtinumus $\log(fea_{prob})$ pagal duotą GMM. Parametrai: požymiai fea, GMM vidurkiai μ, kovariacijos Σ ir svoriai c. Funkcija:

$$\begin{aligned} \log(fea_{prob}) = \; & -\frac{\mathbf{1}}{\mathbf{2}}\left(\sum_{j=1}^{2048}\left(\mu_{jm} \cdot \frac{\mu_{jm}}{\Sigma_{jm}}\right) + \sum_{j=1}^{2048} log(\Sigma_{jm})\right) - \\ & -\frac{\mathbf{1}}{\mathbf{2}}\left(\left(\frac{1}{\Sigma}\right)^{T} \cdot (fea \times fea) - -2\left(\frac{\mu}{\Sigma}\right)^{T} fea + fea_{len} \cdot log(2\pi)\right) + log(c) \end{aligned} \tag{2.18}$$

Čia fea_{len} – balso sričių požymių ilgiai.

Toliau apibūdinamas didžiausio tikėtinumo nustatymo funkcijos veikimo principas. Funkcijos parametrai: svorių koeficientai c, vidurkių vektorius μ ir kovariacijų matrica Σ. Logika: GMM svorių c, vidurkių μ ir kovariacijų Σ įverčiai nustatomi taikant (2.2) išraiškas, nustačius parametrus filtruojama kovariacijų matrica – kviečiama kovariacijų filtravimo funkcija. Po filtravimo modeliui priskiriamos gautos parametrų reikšmės: $\hat{c}, \hat{\mu}, \hat{\Sigma}$ – svorių, vidurkio vektorius ir kovariacijų matricos įverčiai.

Kovariacijų filtravimo funkcija grąžina filtruotą kovariacijų matricą. Įvesties parametrai: $\hat{c}, \hat{\mu}, \hat{\Sigma}$ – svorių, vidurkio vektorius ir kovariacijų matricos įverčiai, kovariacijų matricos slenkstinė riba. Algoritmas: sudaroma slenkstinės ribos matrica $vFloor = \left(\hat{c} \cdot \hat{\Sigma}^T\right) \cdot \text{var\_const}$, atliekamas kovariacijų įvertinių filtravimas: vykdomas kovariacijų $\hat{\Sigma}$ ir slenkstinės ribos $vFloor$ matricų palyginimas, išrenkamos ir grąžinamos maksimalios lyginamų elementų reikšmės.

Aukščiau nurodytomis funkcijomis įgyvendinamas EM algoritmas. Toliau aptarsime GMM modelio bendrinimo funkciją.

GMM modelio bendrinimo funkcijos įvesties parametrai: $\hat{c}_{jm}, \hat{\mu}_{jm}, \hat{\Sigma}_{jm}$ – svorių, vidurkių ir kovariacijų komponentės j būsenoje m įverčiai. GMM mokymas vykdomas 2048 komponentėms ir 60 požymių, todėl $J = 2048$, $M = 60$. Ši funkcija padvigubina modelio parametrų komponenčių skaičių:

$$\hat{c} = \frac{1}{2}\begin{pmatrix} \hat{c}_{11} \\ \hat{c}_{21} \\ \vdots \\ \hat{c}_{J1} \\ \hat{c}_{11} \\ \hat{c}_{21} \\ \vdots \\ \hat{c}_{J1} \end{pmatrix}, \tag{2.19}$$

$$\hat{\mu} = \begin{pmatrix} \hat{\mu}_{11} - \varepsilon_1 & \hat{\mu}_{11} + \varepsilon_1 & \hat{\mu}_{12} - \varepsilon_2 & \hat{\mu}_{12} + \varepsilon_2 & \dots & \hat{\mu}_{1M} - \varepsilon_M & \hat{\mu}_{1M} + \varepsilon_M \\ \hat{\mu}_{21} - \varepsilon_1 & \hat{\mu}_{21} + \varepsilon_1 & \hat{\mu}_{22} - \varepsilon_2 & \hat{\mu}_{22} + \varepsilon_2 & \dots & \hat{\mu}_{2M} - \varepsilon_M & \hat{\mu}_{2M} + \varepsilon_M \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \hat{\mu}_{J1} - \varepsilon_1 & \hat{\mu}_{J1} + \varepsilon_1 & \hat{\mu}_{J2} - \varepsilon_2 & \hat{\mu}_{J2} + \varepsilon_2 & \dots & \hat{\mu}_{JM} - \varepsilon_M & \hat{\mu}_{JM} + \varepsilon_M \end{pmatrix}, \tag{2.20}$$

$$\varepsilon_j = \sqrt{max(\hat{\Sigma}_{j1}, \hat{\Sigma}_{j2}, \dots, \hat{\Sigma}_{jM})}, \tag{2.21}$$

$$\hat{\Sigma} = \begin{pmatrix} \hat{\Sigma}_{11} & \hat{\Sigma}_{1\,2} & \dots & \hat{\Sigma}_{1M} & \hat{\Sigma}_{11} & \hat{\Sigma}_{21} & \dots & \hat{\Sigma}_{1M} \\ \hat{\Sigma}_{21} & \hat{\Sigma}_{22} & \dots & \hat{\Sigma}_{2M} & \hat{\Sigma}_{21} & \hat{\Sigma}_{22} & \dots & \hat{\Sigma}_{2M} \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \hat{\Sigma}_{J1} & \hat{\Sigma}_{J2} & \dots & \hat{\Sigma}_{JM} & \hat{\Sigma}_{J1} & \hat{\Sigma}_{J2} & \dots & \hat{\Sigma}_{JM} \end{pmatrix}. \tag{2.22}$$

Čia $\hat{c}, \hat{\mu}, \hat{\Sigma}$ – svorių, vidurkių, kovariacijų įverčiai, ε_j – j-oji paklaida.

Toliau aptariamas GMM mokymo etapas, kuriuo sukuriamas kalbos signalų akustinis modelis.

2.2.3. GMM mokymas

GMM mokymui, taikant EM algoritmą, vykdoma po 12 modelio tikslinimo ir didinimo iteracijų. Iteracinio skaičiavimo metu kiekviena GMM tikėtinumo įvertinio reikšmė išreiškiama per buvusią reikšmę.

Kalbos signalų akustinio modelio sudarymas GMM metodu realizuojamos algoritmu: nuskaitomi požymiai iš sąrašo *FILES*, nuskaitytiems požymiams nustatomi jų GMM tikėtinumai – kviečiama tikėtinumų nustatymo funkcija, nustatomi tikėtinumų įvertiniai – kviečiama didžiausio tikėtinumo nustatymo funkcija, grąžinamas GMM modelis, modelio parametrai išsaugomi, bendrinamas GMM modelis – kviečiama GMM bendrinimo funkcija, gautos bendrintos GMM modelio reikšmės taikomos kitai iteracijai – su naujomis GMM parametrų reikšmėmis kartojamas EM algoritmas pagal pasirinktą iteracijų skaičių.

Įvykdžius algoritmą visoms modelio tikslinimo ir didinimo iteracijoms grąžinamas ir išsaugomas sudarytas GMM akustinis modelis.

2.3. Akustinio modelio sudarymas taikant k-vidurkių klasterizavimo metodą

Parinkta kita kalbos signalų akustinio modelio sudarymo metodika – akustinio modelio sudarymui taikomas nehierarchinis k-vidurkių (angl. k-mean) klasterizavimo metodas. Metodas laikomas EM algoritmo supaprastinimu. Plačiau apie k-vidurkių klasterizavimo metodą rašoma šaltinyje [34].

Modeliavimas realizuojamas pusprograme *k-vidurkiu_mokymas.py* (žr. 5 priede), o akustinio modelio mokymo logika pateikiama 2.3 paveiksle.

```
Importuojamos bibliotekos
FILES = vykdomas pasirinkto aplanko požymių nuskaitymas
Inicializuojami konfigūracijos parametrai
Pasirenkame centroidų centrus
FOR tikslinimo iteracijos
    FOR požymiai iš sąrašo FILES
        FOR požymių vektoriai laike
            Skaičiuojami atstumai iki centrų
            Išrenkami mažiausi atstumai
            Požymių vektorius pridedamas prie kaupiklio pagal centro numerį
            Padidiname skaitiklį vienetu
        END FOR
    END FOR
    Sukauptų požymių sumą padalinam iš skaitiklių reikšmių
    Rezultatų išsaugojimas
    Perskaičiuojami centroidų centrai
END FOR
```

2.3 pav. Akustinio modelio sudarymo k-vidurkių klasterizavimo metodu pseudokodas

Mokymo realizavimui importuojamos reikiamos bibliotekos (*fnmatch*, *numpy*, *os*), inicializuojami įvesties ir konfigūracijos parametrai. Modelio sukūrimui taikomi moterų ir vyrų įrašų balso sričių MFCC požymiai, todėl sudaromas pasirinkto aplanko požymių sąrašas *FILES*. Be to akustinis modelis sudaromas 2048 komponentėms ir 60 požymių, o mokymo metu vykdoma 21 iteracija.

Akustinio modelio sudarymo k-vidurkių klasterizavimo metodu pseudokodo algoritmas įgyvendinamas 2 etapais: priskyrimo ir atnaujinimo. Priskyrimo etapas laikomas EM algoritmo tikėtinumo žingsniu, o atnaujinimo – maksimizavimo. Prieš etapų vykdymą pagal parinktą komponenčių ir požymių skaičių atsitiktinai sugeneruojami centroidų centrai.

Priskyrimo etape nuskaitinėjami požymiai, paeiliui apskaičiuojami komponenčių atstumai iki centroidų centrų – taikoma kvadratinė Euklido metrika, išrenkami mažiausi atstumai, kaupiama balso sričių požymių suma – požymių vektorius pridedamas prie kaupiklio pagal centro numerį, skaitiklis padidinamas vienetu – skaičiuojama kiek požymių pateko į skaitiklį.

Vykdant atnaujinimo etapą perskaičiuojami centroidų centrai, t. y. sukaupta požymių suma padalinama iš skaitiklių reikšmių. Gauti požymių vidurkiai ir skaitikliai išsaugomi. Naujomis centroidų centrų reikšmėmis perskaičiuojami požymių vidurkiai. Priskyrimo etapas kartojamas pagal parinktą iteracijų skaičių. Įvykdžius visas iteracijas grąžinamas sudarytas akustinis modelis. Toliau aptariama sudarytų akustinių modelių tyrimo logika.

2.4. Modelių tyrimas

Kad būtų nustatytas sudarytų modelių tinkamumas būtinas tyrimas. Dėl to įvertinant modelius išrinktos 6 dažniausiai pasitaikančios modelių komponentės ir atlikta įvairių kalbų įrašų logaritminių tikėtinumų gautų taikant įvairius akustinius modelius statistinė analizė.

2.4.1. Dažniausiai pasitaikančių komponenčių tyrimas

Akustinius modelius sudaro GMM, kurių viršūnės – tai klasterių centrai, o apie juos išsisklaidę požymiai. Akustinio modelio sukūrimas – centrų ir kovariacijų, nulemiančių požymių sklaidą apie tuos centrus, suradimas. Dėl to, siekiant išsiaiškinti ar įvairių akustinių modelių komponenčių požymiai po komponentes pasiskirsto vienodai, atlikta dažniausiai pasitaikančių įvairių akustinių modelių (žr. 2.10 lentelė) komponenčių analizė. Analizės logika pateikiama dažniausiai pasitaikančių komponenčių pusprogramėje *dazniausiai_pasitaikancios_komponentes.py* (žr. 6 priede).

2.10 lentelė. Įvairių kalbų akustinių modelių žymėjimas

Žymėjimas	Akustinis modelis
1	Anglų kalbos GMM modelis
2	Anglų k-vidurkių klasterizavimo modelis
3	Ispanų kalbos GMM modelis
4	Ispanų kalbos k-vidurkių klasterizavimo modelis
5	Italų kalbos GMM modelis
6	Italų kalbos k-vidurkių klasterizavimo modelis
7	Prancūzų kalbos GMM modelis
8	Prancūzų kalbos k-vidurkių klasterizavimo modelis
9	Rusų kalbos GMM modelis
10	Rusų kalbos k-vidurkių klasterizavimo modelis
11	Vokiečių kalbos GMM modelis
12	Vokiečių kalbos k-vidurkių klasterizavimo modelis

Tyrimo metu išrinktos įvairių akustinių modelių dažniausiai pasitaikančios 6 balso ir 6 fono sričių komponentės, nustatytos jų užimamų sričių procentinės išraiškos, t. y. apskaičiuota, kokią balso ar fono sričių dalį įvairių akustinių modelių komponentės užima. Įvertinant modelius atsižvelgiama į kalbančiųjų lytį ir kalbą.

2.4.2. Logaritminių tikėtinumų analizė

Siekiant išsiaiškinti, ar akustiniai modeliai tinkami kalbėtojo atpažinimui, atliktas tyrimas. Tyrimo metu kiekvienai tirtai kalbai parinkta po 10 testavimo įrašų, iš kurių: 5 vyrų ir 5 moterų. Nuskaičius įrašus apskaičiuoti jų logaritminiai tikėtinumai taikant įvairių kalbų akustinius modelius (žr. 2.10 lentelė). Anglų, ispanų, italų, prancūzų, rusų ir vokiečių kalbų garso įrašų logaritminių tikėtinumų nustatymo algoritmas realizuojamas 7 priedo pusprograme *logaritminiai_tiketinumai.py*. Pagal gautus įvairių kalbų garso įrašų logaritminių tikėtinumų rezultatus atlikta statistinė analizė, kurios metu atlikta vieno veiksnio dispersinė analizė (angl. one-way ANOVA) ir Stjudento t-testas (angl. Student t Test). Statistinė analizė įgyvendinama pusprograme *statistiniai_testai.py* (žr. 8 priedas).

Vieno veiksnio dispersinės analizės metu veiksniu parenkama kalba, o reikšmingumo lygmuo 0,05. Šios analizės metu mėginta išsiaiškinti, ar logaritminių tikėtinumų rezultatai nuo kalbos statistiškai reikšmingai skiriasi. Atliekant Stjudento t-testą tikrinta, ar anglų kalbos akustinių modelių logaritminiai tikėtinumai statistiškai reikšmingai skiriasi nuo lyginamų kalbų esant reikšmingumo lygmeniui 0,05. Pagal gautus rezultatus padarytos statistinės išvados.

Be to atlikta 1-ojo ir 2-ojo akustinių modelių logaritminių tikėtinumų vidurkių analizė, kurios metu siekta išsiaiškinti, kuri tiriama kalba labiausiai skiriasi nuo anglų kalbos. Dėl to gautos vidurkių reikšmės išrikiuotos nuo mažiausios iki didžiausios. Išrinkus skirtingiausią kalbą atsitiktinai parinkta

dar po 100 tos kalbos ir anglų kalbos testavimo įrašų, taikant anglų kalbos akustinius modelius apskaičiuoti jų logaritminiai tikėtinumai. Pagal gautus rezultatus atliktas Stjudento t-testas, kurio metu siekta išsiaiškinti, ar lyginamų kalbų įrašų logaritminiai tikėtinumai statistiškai reikšmingai skiriasi, taikant anglų kalbos akustinius modelius. Čia lyginamos kalbos – anglų ir kalba, kuri labiausiai skiriasi nuo anglų kalbos, o reikšmingumo lygmuo – 0,05. Pagal gautas statistinio testo p reikšmes padarytos išvados.

3. KALBOS SIGNALŲ AKUSTINIŲ MODELIŲ, SKIRTŲ KALBĖTOJO ATPAŽINIMUI, SUDARYMO REZULTATAI

3.1. MFCC požymių išskyrimo rezultatai

Požymių išskyrimui taikyti anglų, prancūzų, rusų, ispanų, italų ir vokiečių kalbų garso įrašai. Bendrumo dėlei parinkti įvairių triukšmingumo lygio aplinkų, skirtingo amžiaus ir lyčių asmenų garso įrašai.

MFCC požymių išskyrimui taikyta komponenčių statistikos tyrimo pseudokodo (žr. 2.1 pav.) logika: kad balso sričių aptikimo detektorius balso sritims nepriskirtų fono sričių, ieškant komponenčių atkreipiamas dėmesys į požymių normavimą, kurio metu atskiriami 0-inis ir 3-iasis MFCC požymiai bei normuojami pagal failo duomenis, o likę požymiai normuojami pagal statistinius duomenis. Po normavimo požymiai sustatomi ankstesniąja tvarka ir sugrupuojami į balso ir fono sričių požymius. Be to atskirtos vyrų ir moterų įrašų komponentės.

Siekiant nustatyti gautų rezultatų tinkamumą – būtina analizė. Šiam tikslui pasiekti apskaičiuotos komponenčių statistikos, sudaryti komponenčių pasikartojimo dažnių įrašuose pasiskirstymo grafikai. Kiekvienai tirtai kalbai akustinio modelio komponenčių rezultatai vaizduojami atskirai.

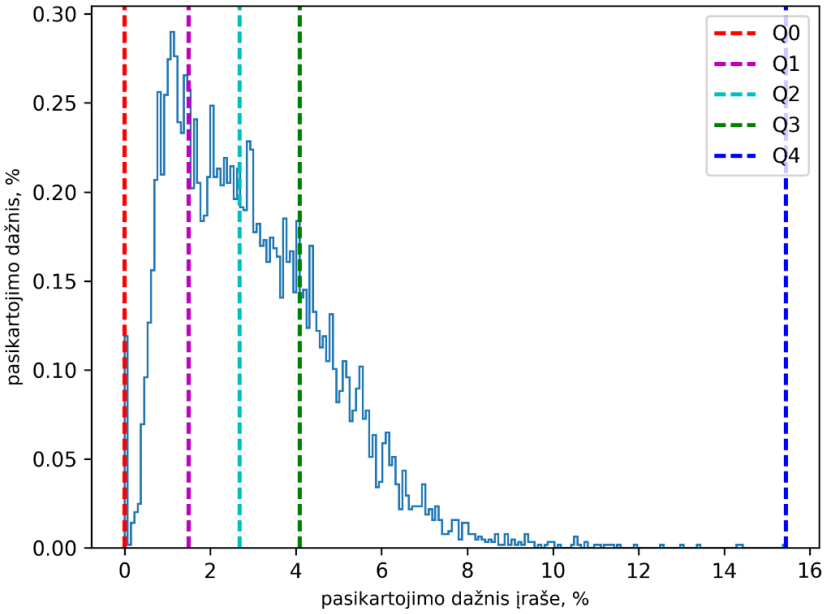
Sudaroma po 3 moterų ir vyrų, fono ir balso sričių komponenčių pasikartojimo dažnių įrašuose pasiskirstymo grafikus, kuriuose vaizduojami geriausi rezultatai medianos atžvilgiu.

Siekiant išsiaiškinti, ar kalbos požymiai po komponentes pasiskirto vienodai, atlikta dominuojančių komponenčių analizė, kurios metu, taikant įvairių kalbų akustinius modelius, nustatytos dažniausiai pasikartojančios tų kalbų garso įrašų balso ir fono sričių komponentės, apskaičiuotos jų užimamų sričių dalys. Gauti rezultatai suvesti į lenteles ir atlikta jų analizė.

3.1.1. Anglų kalbos garso įrašų rezultatai

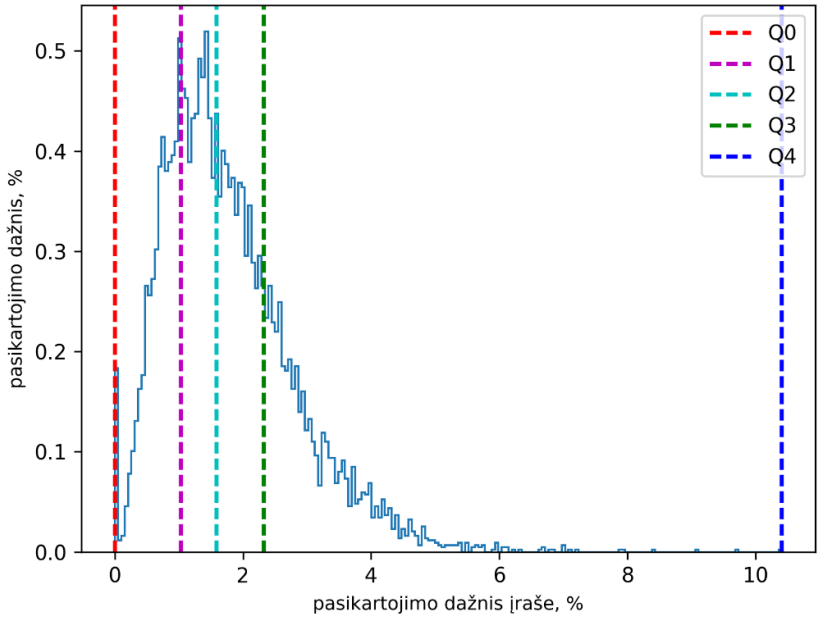
Siekiant išskirti kalbos požymius ir apskaičiuoti modelių komponenčių statistikas tirti 14876 anglų kalbos telefoninių pokalbių įrašai, iš kurių: 8395 parinkti moterų ir 6481 – vyrų. Pirma aptariami moterų garso įrašų modelio komponenčių rezultatai.

Moterų balso sričių 1-ojo modelio komponenčių 548, 1060 ir 287 pasikartojimo dažnių įrašuose pasiskirstymai pateikiami 3.1, 3.2 ir 3.3 paveiksluose.



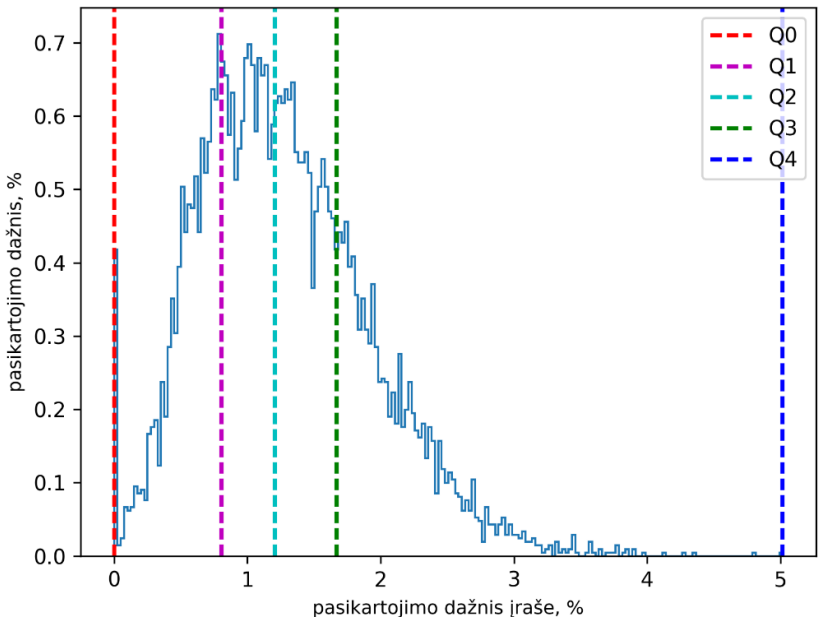
3.1 pav. Moterų balso srities 1-ojo modelio komponentės 548 pasikartojimo dažnių įrašuose pasiskirstymas

Išanalizavus moterų balso srities 1-ojo modelio komponentės 548 pasikartojimo dažnių įrašuose pasiskirstymą pastebėta, kad komponentės pirmojo kvartilio reikšmė – tik 1,499%, antrojo – 2,681%, trečiojo – 4,088%.



3.2 pav. Moterų balso srities 1-ojo modelio komponentės 1060 pasikartojimo dažnių įrašuose pasiskirstymas

Nustatyta, kad moterų balso srities 1-ojo modelio komponentės 1060 pasikartojimo dažnių įrašuose pirmojo kvartilio reikšmė – 1,031%, antrojo – 1,588%, trečiojo – 2,325%.

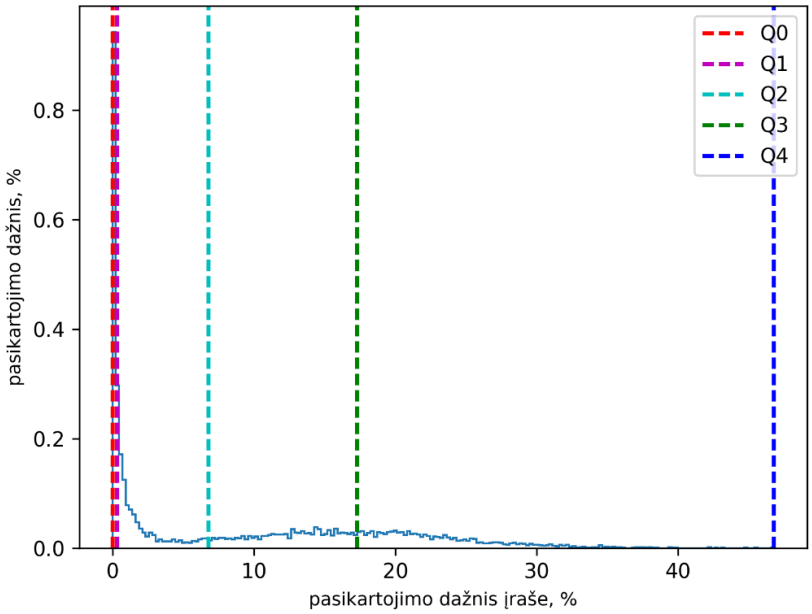


3.3 pav. Moterų balso srities 1-ojo modelio komponentės 287 pasikartojimo dažnių įrašuose pasiskirstymas

Moterų balso srities 1-ojo modelio komponentės 287 pasikartojimo dažnių įrašuose pirmojo kvartilio reikšmė – tik 0,807%, medianos – 1,207%, trečiojo kvartilio – 1,669%.

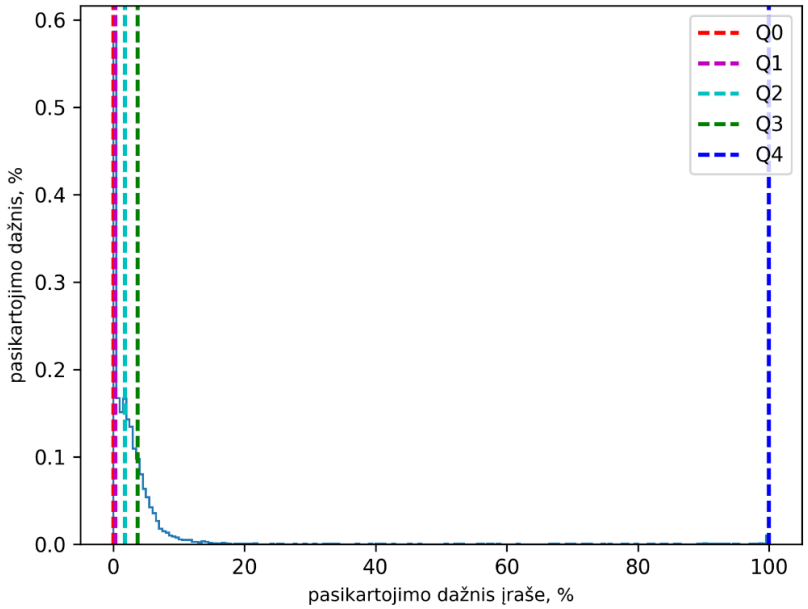
Atlikus tyrimą pastebėta, kad dažniausiai pasitaikančios 6 moterų balso sričių 1-ojo modelio komponentės (548, 1060, 676, 1188, 1700, 164) užima 12,5%, o 2-ojo modelio komponentės (566, 210, 804, 326, 1250, 379) – 2,2% moterų įrašų balso sričių.

Apačioje pateikiami moterų fono sričių 1-ojo modelio komponenčių 1865, 9 ir 1700 grafikai.



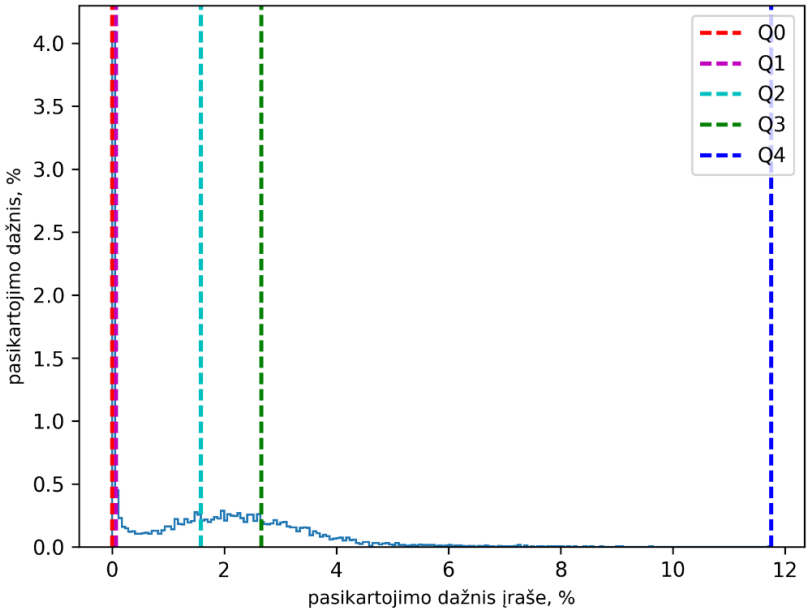
3.4 pav. Moterų fono srities 1-ojo modelio komponentės 1865 pasikartojimo dažnių įrašuose pasiskirstymas

Moterų fono srities 1-ojo modelio komponentės 1865 pasikartojimo dažnių įrašuose (žr. 3.4 pav.) pirmojo kvartilio reikšmė 0,319%, antrojo kvartilio – 6,803%, trečiojo kvartilio – 17,304%.



3.5 pav. Moterų fono srities 1-ojo modelio komponentės 9 pasikartojimo dažnių įrašuose pasiskirstymas

Ištyrus moterų fono srities 1-ojo modelio komponentę 9 (žr. 3.5 pav.) nustatyta, kad jos pasikartojimo dažnių įrašuose pirmojo kvartilio reikšmė – 0,307%, antrojo – 1,774%, trečiojo – 3,712%.

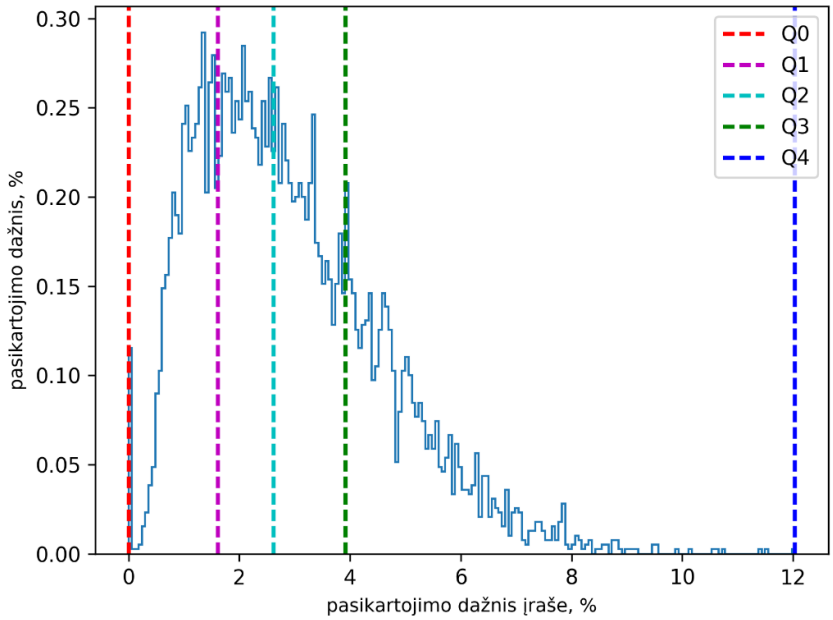


3.6 pav. Moterų fono srities 1-ojo modelio komponentės 1700 pasikartojimo dažnių įrašuose pasiskirstymas

Moterų fono srities 1-ojo modelio komponentės 1700 pasikartojimo dažnių įrašuose (žr. 3.6 pav.) pirmojo kvartilio reikšmė – 0,075%, antrojo – 1,587%, trečiojo – 2,661%.

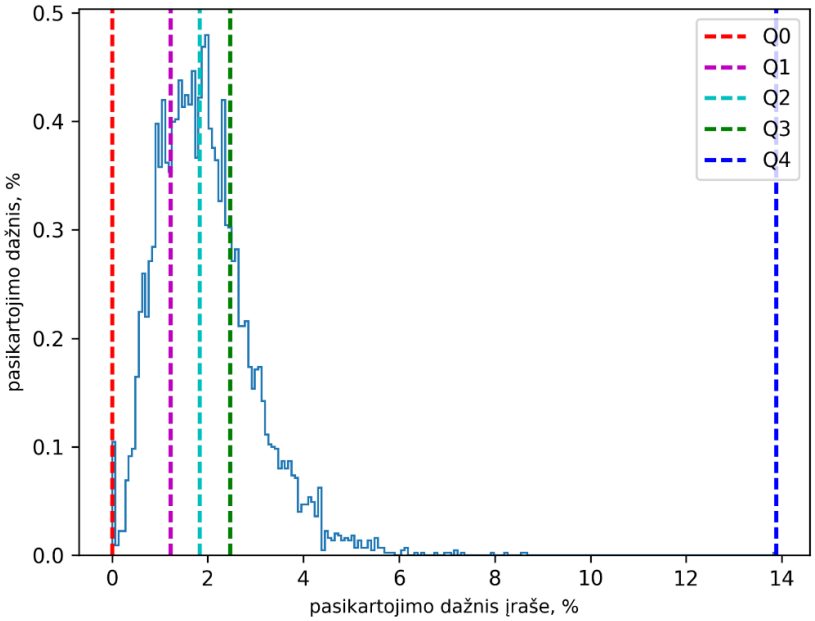
Nustatyta, kad moterų fono srityse dominuojančios 1-ojo modelio komponentės (201, 1662, 505, 337, 1781, 1609) užima 7,1%, o dažniausiai pasikartojančios 2-ojo modelio komponentės (1624, 566, 438, 1805, 486, 1871) užima 8,1% moterų įrašų fono sričių.

Apačioje 3.7, 3.8 ir 3.9 paveiksluose vaizduojami vyrų balso sričių 1-ojo modelio komponenčių 548, 1060 ir 287 pasikartojimo dažnių įrašuose pasiskirstymai.



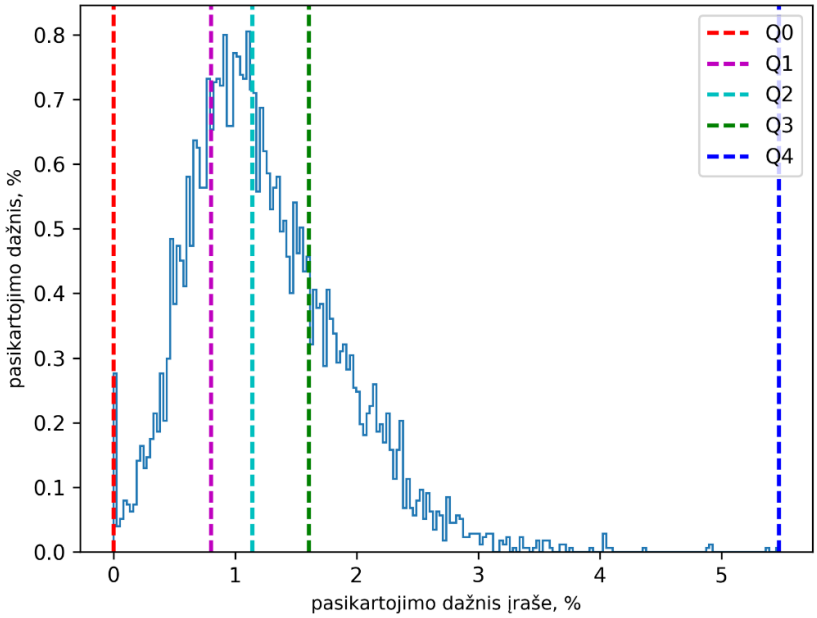
3.7 pav. Vyrų balso srities 1-ojo modelio komponentės 548 pasikartojimo dažnių įrašuose pasiskirstymas

Vyrų balso srities 1-ojo modelio komponentės 548 pasikartojimo dažnių įrašuose pirmasis kvartilis siekia tik 1,613%, antrasis – 2,620%, trečiasis – 3,923%.



3.8 pav. Vyrų balso srities 1-ojo modelio komponentės 1060 pasikartojimo dažnių įrašuose pasiskirstymas

Vyrų balso srities 1-ojo modelio komponentės 1060 pasikartojimo dažnių įrašuose pirmojo kvartilio reikšmė – 1,229%, antrojo kvartilio – 1,836%, trečiojo – 2,471%.

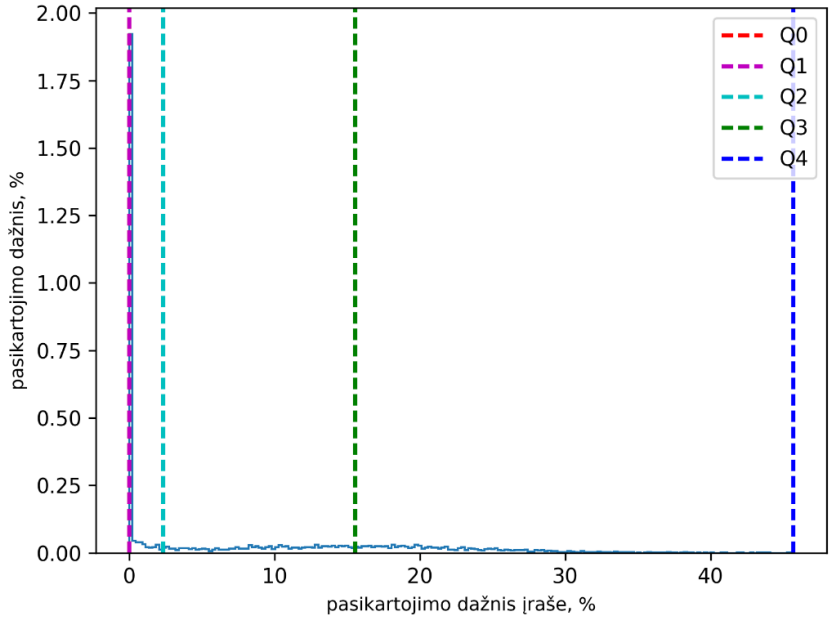


3.9 pav. Vyrų balso srities 1-ojo modelio komponentės 287 pasikartojimo dažnių įrašuose pasiskirstymas

Vyrų balso srities 1-ojo modelio komponentės 287 pasikartojimo dažnių įrašuose pirmojo kvartilio reikšmė siekia vos 0,803%, antrojo – 1,144%, trečiojo – 1,607%.

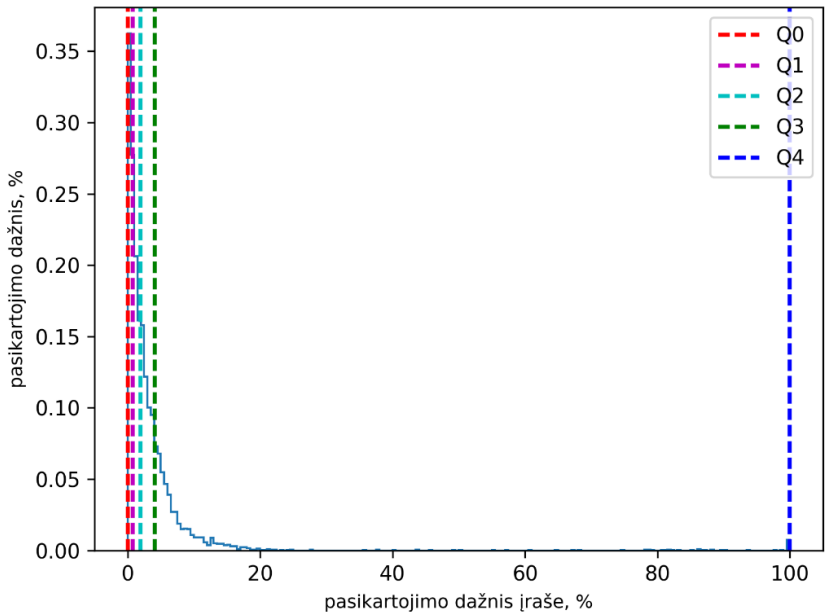
Vyraujančios vyrų balso sričių 1-ojo modelio komponentės (548, 1060, 676, 1311, 164, 1700) užima 9% vyrų įrašų balso sričių, 2-ojo modelio (1959, 1809, 1822, 397, 118, 1993) – 1,9%.

Žemiau pateikiami vyrų fono sričių 1-ojo modelio komponenčių 1865, 9 ir 1700 rezultatai.



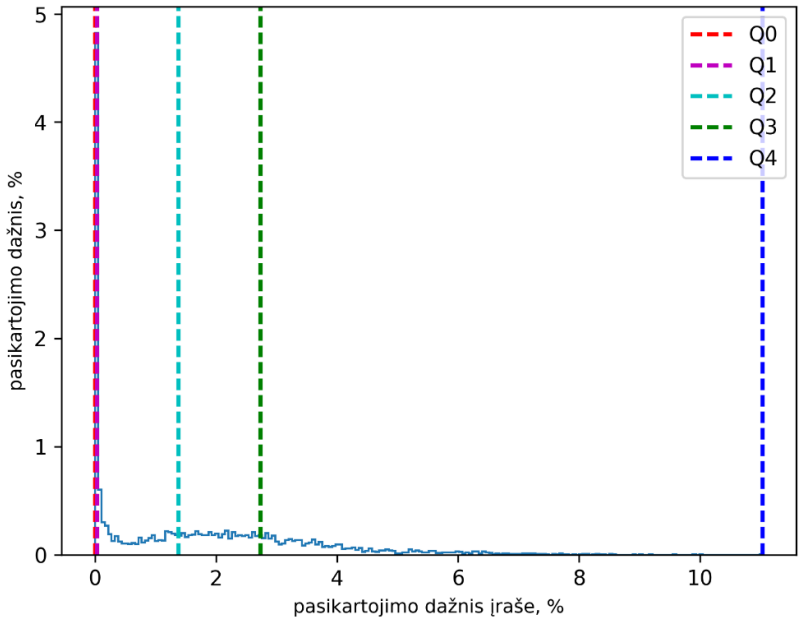
3.10 pav. Vyrų fono srities 1-ojo modelio komponentės 1865 pasikartojimo dažnių įrašuose pasiskirstymas

Vyrų fono srities 1-ojo modelio komponentės 1865 pasikartojimo dažnių įrašuose (žr. 3.10 pav.) pirmojo kvartilio reikšmė – 0%, antrojo – 2,356%, trečiojo – 15,564%.



3.11 pav. Vyrų fono srities 1-ojo modelio komponentės 9 pasikartojimo dažnių įrašuose pasiskirstymas

Vyrų fono srities 1-ojo modelio komponentės 9 pasikartojimo dažnių įrašuose (žr. 3.11 pav.) pirmojo kvartilio reikšmė – 0,733%, mediana siekia 1,976%, o trečiojo kvartilio reikšmė – 4,101%.



3.12 pav. Vyrų fono srities 1-ojo modelio komponentės 1700 pasikartojimo dažnių įrašuose pasiskirstymas

Vyrų fono srities 1-ojo modelio komponentės 1700 pasikartojimo dažnių įrašuose (žr. 3.12 pav.) pirmojo kvartilio reikšmė siekia tik 0,038%, medianos – 1,385%, o trečiojo – 2,738%.

Pastebėta, kad vyrų fono srityse dominuojančios 1-ojo modelio komponentės (1060, 676, 1700, 548, 36, 1572) užima 15,7%, o dažniausiai pasitaikančios 2-ojo modelio komponentės (597, 820, 1479, 669, 118, 1545) užima 4,9% vyrų įrašų fono sričių.

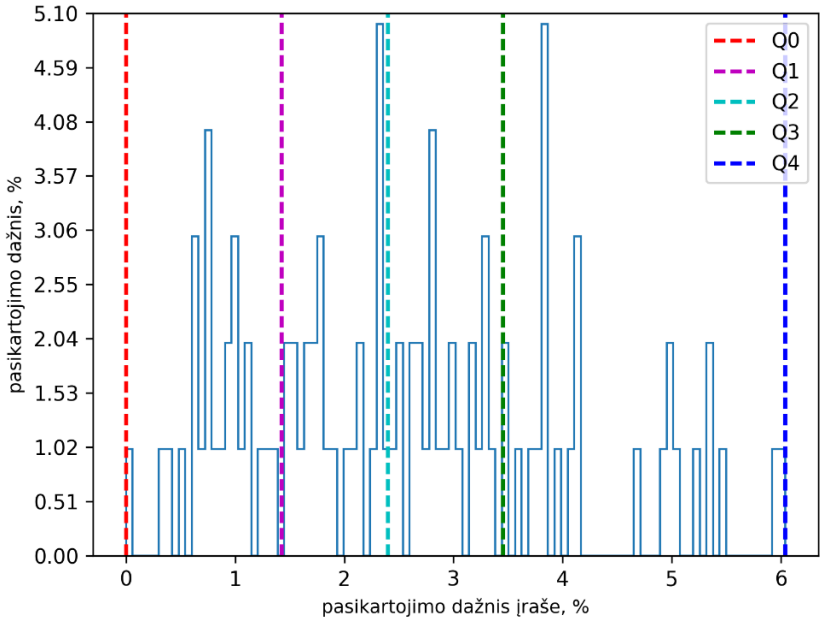
Išnagrinėjus rezultatus nustatyta, kad dažniausiai pasitaikančios moterų ir vyrų garso įrašų fono sričių 1-ojo modelio komponentės medianos atžvilgiu yra tokios pat.

Anglų kalbos įrašų analizė atskleidė, kad fono sričių komponentės skiriasi nuo balso sričių komponenčių. Toliau tikrinama, ar kitų kalbų įrašų balso sričių 1-ojo modelio komponentės taip pat skiriasi nuo fono sričių 1-ojo modelio komponenčių.

3.1.2. Ispanų kalbos garso įrašų rezultatai

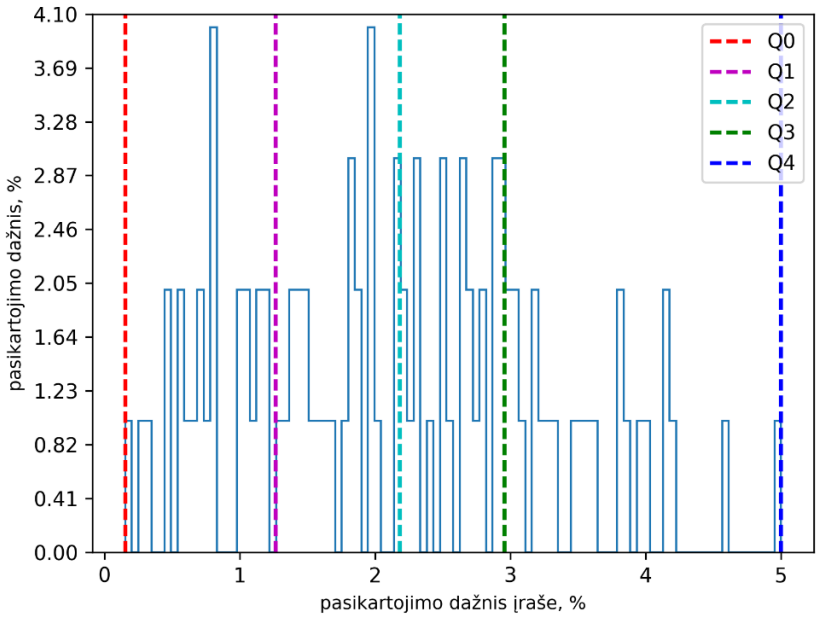
Siekiant išsiaiškinti ar ir ispanų įrašų fono sričių komponentės skiriasi nuo balso sričių komponenčių atliktas tyrimas. Požymių išskyrimui parinkta po 100 moterų ir vyrų ispanų kalbos garso įrašų.

Moterų garso įrašų tyrimo metu išskirstos balso ir fono sričių 1-ojo modelio komponentės. Moterų balso sričių 1-ojo modelio komponenčių 1060, 548 ir 1420 pasikartojimo dažnių įrašuose pasiskirstymai vaizduojami 3.13, 3.14 ir 3.15 paveiksluose.



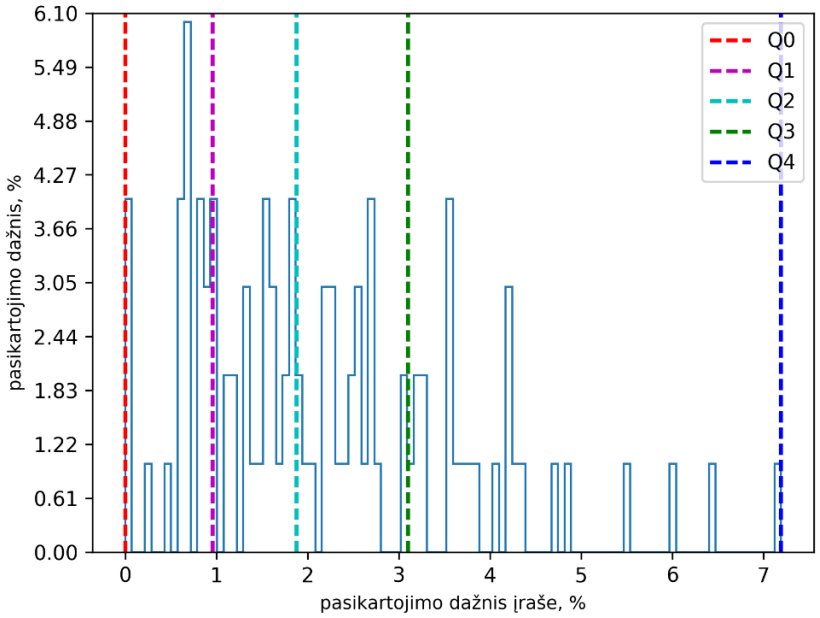
3.13 pav. Moterų balso srities 1-ojo modelio komponentės 1060 pasikartojimo dažnių įrašuose pasiskirstymas

Moterų balso srities 1-ojo modelio komponentės 1060 pasikartojimo dažnių įrašuose pasiskirstymo rezultatai atskleidė, jog komponentės pirmojo kvartilio reikšmė – 1,428%, antrojo kvartilio – 2,401%, trečiojo – 3,453%.



3.14 pav. Moterų balso srities 1-ojo modelio komponentės 548 pasikartojimo dažnių įrašuose pasiskirstymas

Remiantis 3.14 paveikslo rezultatais nustatyta, kad moterų balso srities 1-ojo modelio komponentės 548 pasikartojimo dažnių įrašuose pirmojo kvartilio reikšmė – 1,264%, medianos – 5%, trečiojo kvartilio – 2,956%.



3.15 pav. Moterų balso srities 1-ojo modelio komponentės 1420 pasikartojimo dažnių įrašuose pasiskirstymas

Išanalizavus 3.15 paveiksle pateiktą moterų balso srities 1-ojo modelio komponentės 1420 pasikartojimo dažnių įrašuose pasiskirstymą išsiaiškinta, kad komponentės pirmojo kvartilio reikšmė siekia tik 0,961%, medianos – 1,88%, trečiojo kvartilio – 3,101%.

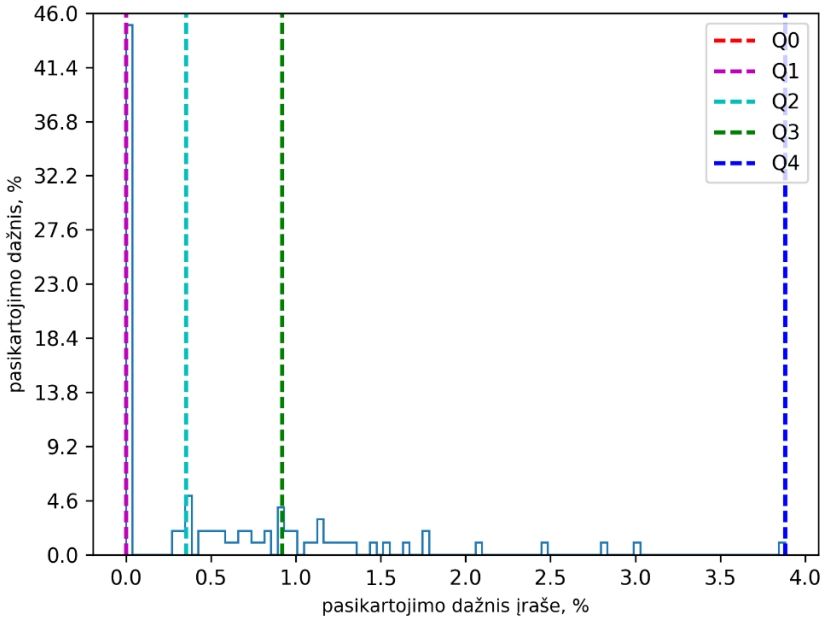
Nustatytos dažniausiai pasitaikančių moterų balso sričių įvairių modelių komponentės. Rezultatai suvesti 3.1 lentelėje.

3.1 lentelė. Ispanų kalbos moterų įrašuose dažniausiai pasitaikančios balso sričių komponentės

Modelis	Balso sričių komponentės	Užima moterų įrašų balso sričių, %
1	1420, 151, 1444, 772, 87, 1022	11,3
2	1335, 822, 1226, 1040, 1912, 910	7,8
3	195, 383, 881, 261, 869, 101	19,3
4	1803, 1547, 1984, 1580, 1923, 1480	7

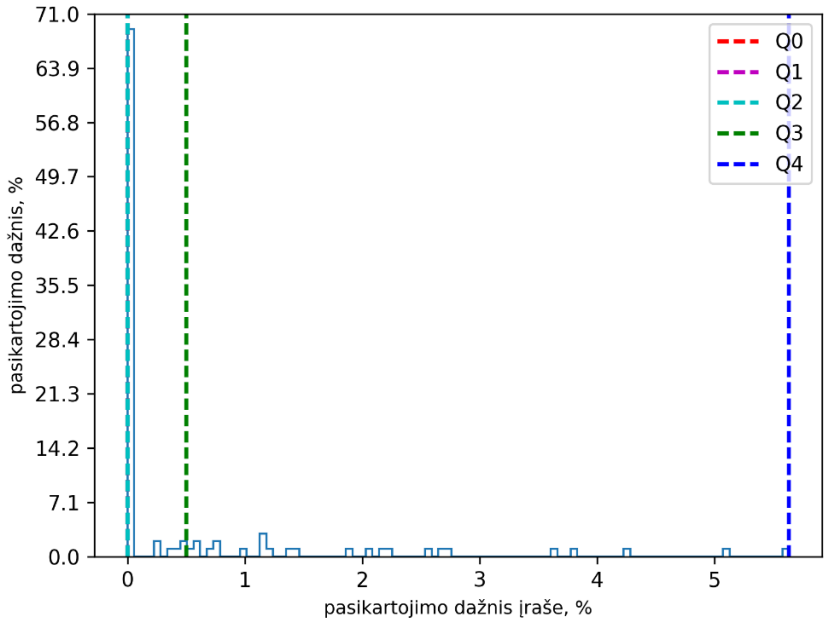
Išsiaiškinta, kad dažniausiai pasitaikančios moterų balso sričių 4-ojo modelio komponentės užima mažiausiai moterų įrašų balso sričių – 7%, o daugiausiai įrašų balso sričių užima 3-ojo modelio balso sričių komponentės – 19,3%.

Moterų fono sričių 1-ojo modelio komponenčių 899, 2046 ir 661 pasikartojimo dažnių įrašuose pasiskirstymai pateikiami 3.16, 3.17 ir 3.18 paveiksluose.



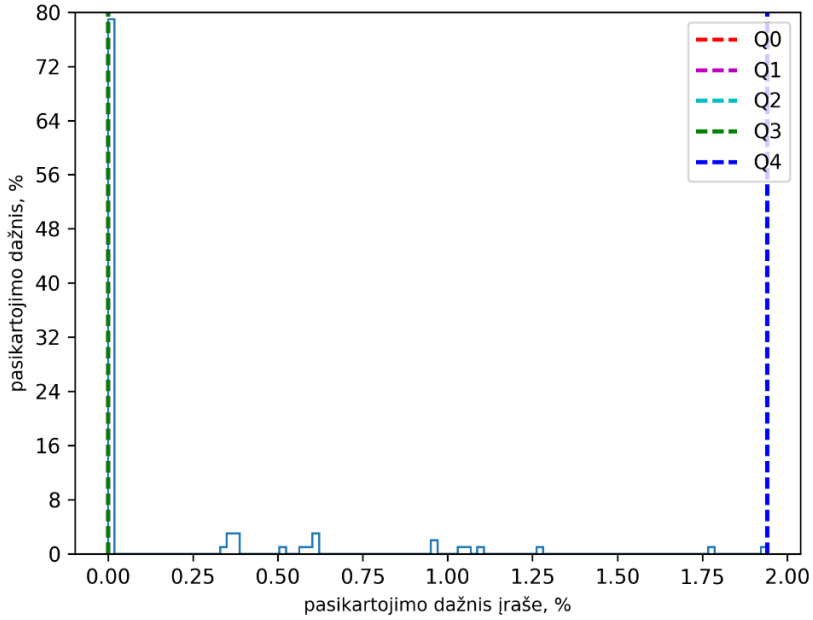
3.16 pav. Moterų fono srities 1-ojo modelio komponentės 899 pasikartojimo dažnių įrašuose pasiskirstymas

Išanalizavus moterų fono srities 1-ojo modelio komponentės 899 pasikartojimo dažnių įrašuose pasiskirstymą (žr. 3.16 pav.) pastebėta, kad komponentės pirmojo kvartilio reikšmė – 0%, antrojo kvartilio – 0,355%, trečiojo – 0,920%.



3.17 pav. Moterų fono srities 1-ojo modelio komponentės 2046 pasikartojimo dažnių įrašuose pasiskirstymas

Ištyrus moterų fono srities 1-ojo modelio komponentės 2046 pasikartojimo dažnių įrašuose pasiskirstymą (žr. 3.17 pav.) nustatyta, kad komponentės pirmojo ir antrojo kvartilių reikšmės tokios pat – po 0%, o trečiojo kvartilio reikšmė siekia tik 0,503%.



3.18 pav. Moterų fono srities 1-ojo modelio komponentės 661 pasikartojimo dažnių įrašuose pasiskirstymas

Ištyrus moterų fono srities 1-ojo modelio komponentės 661 pasikartojimo dažnių įrašuose pasiskirstymą nustatyta, kad komponentės pirmojo, antrojo ir trečiojo kvartilių reikšmės tokios pat – 0%.

Nustatytos ispanų kalbos moterų garso įrašų fono srityse dažniausiai pasitaikančios 6 komponentės (žr. 3.2 lentelė).

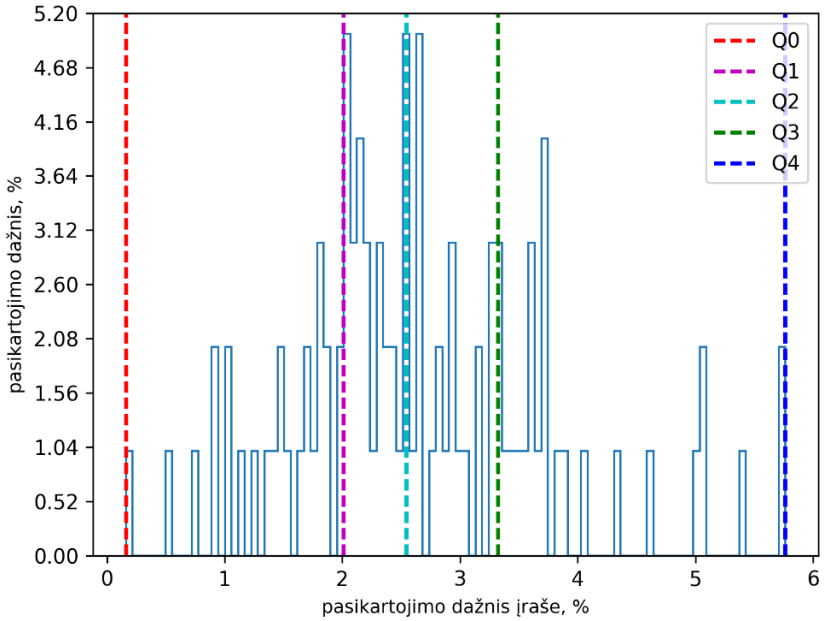
3.2 lentelė. Ispanų kalbos moterų įrašuose dažniausiai pasitaikančios fono sričių komponentės

Modelis	Fono sričių komponentės	Užima moterų įrašų fono sričių, %
1	1758, 510, 1022, 1278, 254, 1534	22,8
2	910, 1821, 1849, 648, 1455, 937	24,6
3	237, 333, 369, 945, 881, 801	43,4
4	1478, 1825, 1981, 1910, 544, 1110	25,9

Gauta, kad moterų garso įrašų fono srityse dominuojančios 6 moterų fono sričių komponentės užima mažiausiai moterų įrašų fono sričių taikant 1-ąjį modelį – 22,8%, o daugiausiai moterų įrašų fono sričių užima taikant 3-iąjį modelį – 43,4%.

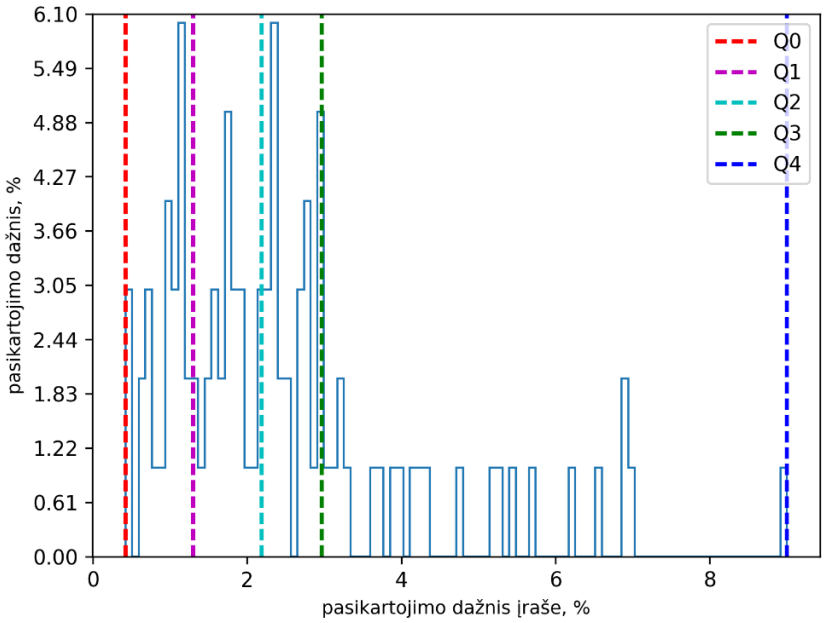
Moterų garso įrašų tyrimo analizė atskleidė, kad dažniausiai pasitaikančios moterų balso sričių komponentės skiriasi nuo moterų fono sričių komponenčių. Toliau analizuojami vyrų įrašų rezultatai.

Vyrų balso sričių 1-ojo modelio komponenčių 1060, 548 ir 676 pasikartojimo dažnių įrašuose pasiskirstymai pateikiami 3.19, 3.20 ir 3.21 paveiksluose.



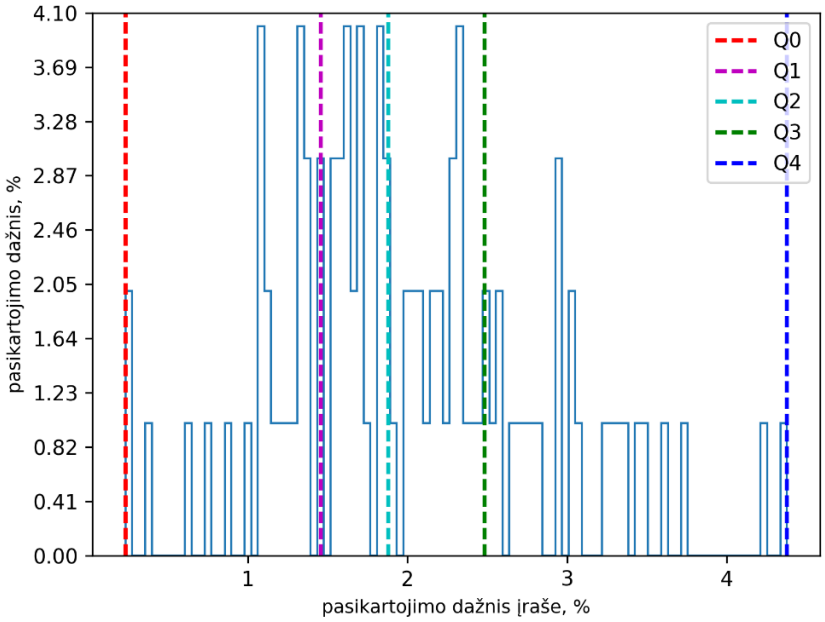
3.19 pav. Vyrų balso srities 1-ojo modelio komponentės 1060 pasikartojimo dažnių įrašuose pasiskirstymas

Išnagrinėjus vyrų balso srities 1-ojo modelio komponentės 1060 pasikartojimo dažnių įrašuose pasiskirstymą (žr. 3.19 pav.) nustatyta, kad komponentės pirmojo kvartilio reikšmė siekia 2,009%, medianos – 2,545%, trečiojo kvartilio – 3,323%.



3.20 pav. Vyrų balso srities 1-ojo modelio komponentės 548 pasikartojimo dažnių įrašuose pasiskirstymas

Ištyrus vyrų balso srities 1-ojo modelio komponentės 548 pasikartojimo dažnių įrašuose pasiskirstymą (žr. 3.20 pav.) išsiaiškinta, kad modelio komponentės pirmojo kvartilio reikšmė siekia tik 1,297%, antrojo kvartilio reikšmė – 2,19%, o trečiojo kvartilio – 2,969%.



3.21 pav. Vyrų balso srities 1-ojo modelio komponentės 676 pasikartojimo dažnių įrašuose pasiskirstymas

Išanalizavus vyrų balso srities 1-ojo modelio komponentės 676 pasikartojimo dažnių įrašuose pasiskirstymą (žr. 3.21 pav.) nustatyta, kad šios komponentės pirmojo kvartilio reikšmė siekia 1,455%, medianos reikšmė siekia 1,878%, trečiojo kvartilio reikšmė – 2,484%.

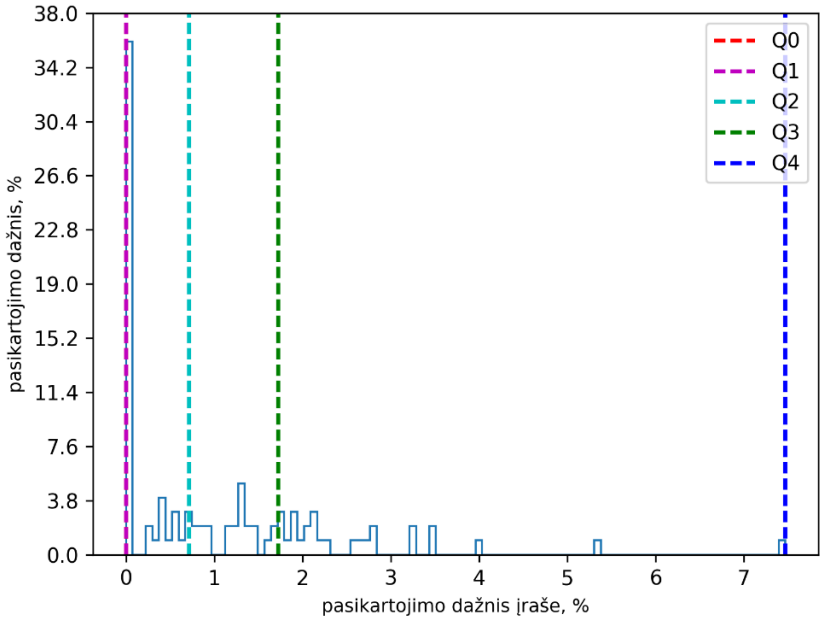
Ištyrus vyrų garso įrašus nustatytos dažniausiai pasitaikančių vyrų balso sričių įvairių akustinių modelių komponenčių užimamos dalys (žr. 3.3 lentelė).

3.3 lentelė. Ispanų kalbos vyrų įrašuose dažniausiai pasitaikančios balso sričių komponentės

Modelis	Balso sričių komponentės	Užima vyrų įrašų balso sričių, %
1	1188, 932, 244, 1060, 1700, 548	11,4
2	3, 77, 773, 793, 807, 1615	5,8
3	261, 55, 23, 375, 39, 167	22,4
4	241, 139, 1129, 1605, 1691, 1682	6,8

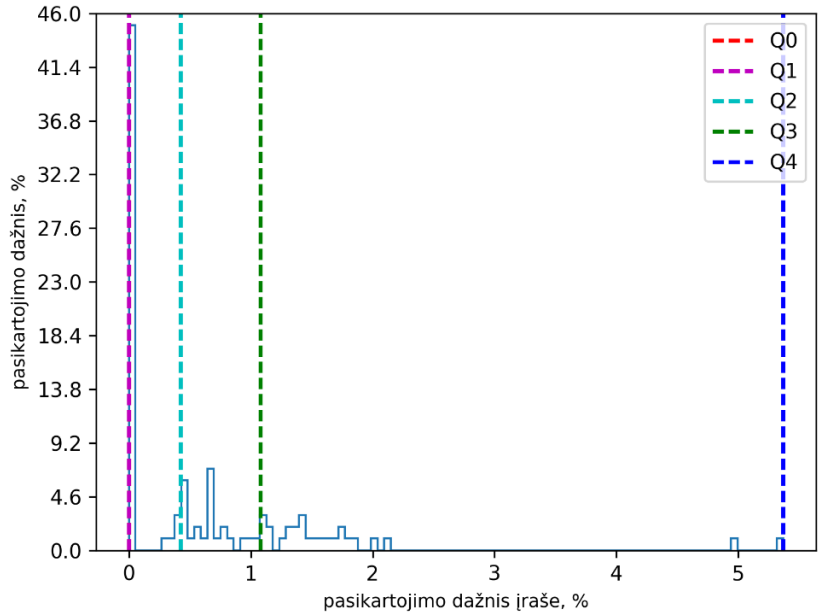
Pagal lentelės rezultatus išsiaiškinta, kad ispaniškai kalbančių vyrų įrašų balso srityse dominuojančios 6 vyrų balso sričių 2-ojo modelio komponentės užima mažiausiai (5,8%), o daugiausiai – 4-ojo modelio komponentės (22,4%).

Vyrų garso įrašų fono sričių 1-ojo modelio komponenčių 613, 244 ir 1202 pasikartojimo dažnių įrašuose pasiskirstymai vaizduojami 3.22, 3.23 ir 3.24 paveiksluose atitinkamai.



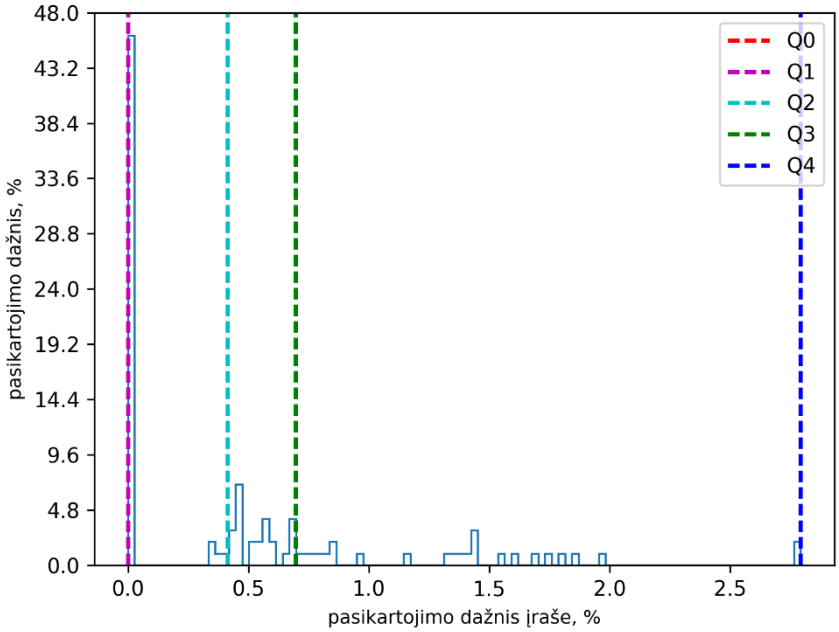
3.22 pav. Vyrų fono srities 1-ojo modelio komponentės 613 pasikartojimo dažnių įrašuose pasiskirstymas

Išanalizavus 3.22 paveiksle vaizduojamą vyrų fono srities 1-ojo modelio komponentės 613 pasikartojimo dažnių įrašuose pasiskirstymą išsiaiškinta, kad komponentės pasikartojimo dažnių įrašuose pirmojo kvartilio reikšmė – 0%, antrojo – 0,717%, trečiojo – 1,728%.



3.23 pav. Vyrų fono srities 1-ojo modelio komponentės 244 pasikartojimo dažnių įrašuose pasiskirstymas

Ištyrus 3.23 paveikslą pastebėta, jog vyrų fono srities 1-ojo modelio komponentės 244 pasikartojimo dažnių ispanų kalbos įrašuose pirmojo kvartilio reikšmė – 0%, medianos reikšmė – 0,426%, trečiojo kvartilio reikšmė – 1,080%.



3.24 pav. Vyrų fono srities 1-ojo modelio komponentės 1202 pasikartojimo dažnių įrašuose pasiskirstymas

Vyrų fono srities 1-ojo modelio komponentės 1202 pasikartojimo dažnių įrašuose pirmojo kvartilio reikšmė – 0%, antrojo kvartilio – 0,413%, trečiojo – 0,697%.

Nustatytos vyrų ispanų kalbos įrašuose dominuojančios fono sričių įvairių akustinių modelių komponentės (žr. 3.4 lentelė).

3.4 lentelė. Ispanų kalbos vyrų įrašuose dažniausiai pasitaikančios fono sričių komponentės

Modelis	Fono sričių komponentės	Užima vyrų įrašų fono sričių, %
1	9, 1545, 1393, 1713, 1633, 281	20,6
2	110, 1201, 47, 1036, 1472, 918	19,4
3	7, 647, 787, 355, 419, 515	38,9
4	686, 1129, 115, 429, 896, 277	30,3

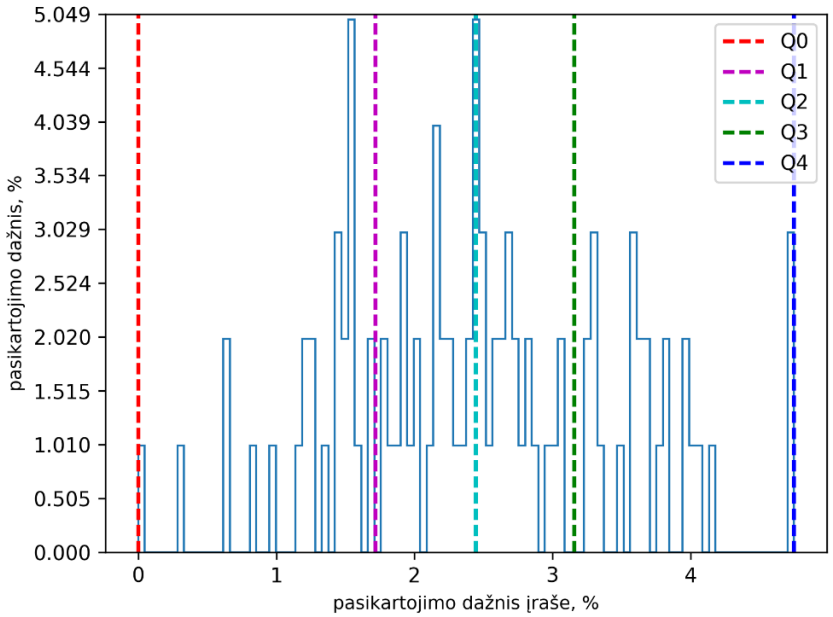
Išsiaiškinta, kad ispanų kalbos vyrų garso įrašuose dominuojančios 6 vyrų fono sričių 3-iojo modelio komponentės užima daugiausia vyrų įrašų fono sričių (38,9%). Mažiausiai vyrų įrašų fono sričių užima 2-ojo modelio komponentės (19,4%).

Vadinasi, ispanų kalbos įrašų požymiai po įvairių akustinių modelių komponentes pasiskirsto nevienodai. Be to ispanų kalbos įrašų balso ir fono sričių 1-ojo modelio komponentės skiriasi.

3.1.3. Italų kalbos garso įrašų rezultatai

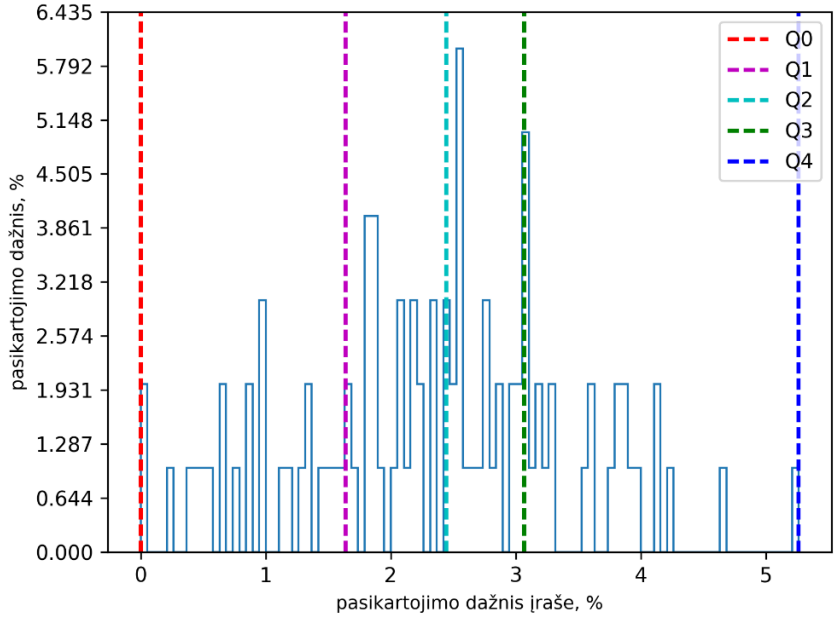
Siekiant pagrindinio darbo tikslo išskirti italų kalba parinktų garso įrašų požymiai, nustatytos modelių komponenčių statistikos. Dėl to parinkta 200 įrašų, iš kurių: 100 – moterų ir 100 – vyrų.

Sudaryti moterų balso sričių 1-ojo modelio komponenčių 548 (žr. 3.25 pav.), 1060 (žr. 3.26 pav.) ir 676 (žr. 3.27 pav.) pasikartojimo dažnių įrašuose pasiskirstymai.



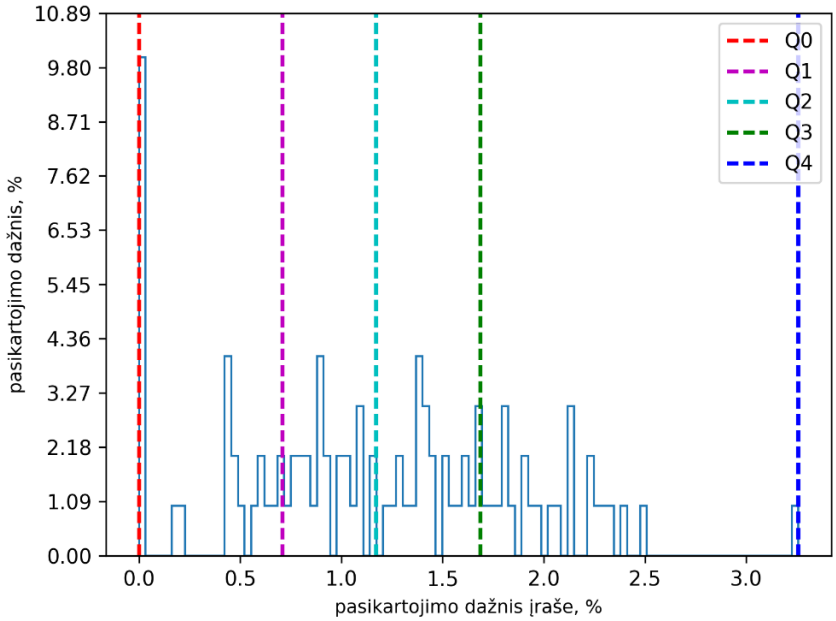
3.25 pav. Moterų balso srities 1-ojo modelio komponentės 548 pasikartojimo dažnių įrašuose pasiskirstymas

Moterų balso srities 1-ojo modelio komponentės 548 pasikartojimo dažnių įrašuose pasiskirstymas (žr. 3.25 pav.) atskleidė, kad komponentės pasikartojimo dažnių įrašuose pirmojo kvartilio reikšmė siekia 1,716%, antrojo – 2,444%, trečiojo – 3,156%.



3.26 pav. Moterų balso srities 1-ojo modelio komponentės 1060 pasikartojimo dažnių įrašuose pasiskirstymas

Išnagrinėjus moterų balso srities 1-ojo modelio komponentės 1060 pasikartojimo dažnių italų kalbos įrašuose pasiskirstymą (žr. 3.26 pav.) nustatyta, jog komponentės pirmojo kvartilio reikšmė siekia 1,640%, antrojo – 2,444%, trečiojo – 3,067%.



3.27 pav. Moterų balso srities 1-ojo modelio komponentės 676 pasikartojimo dažnių įrašuose pasiskirstymas

Analizuojant moterų balso srities 1-ojo modelio komponentės 676 pasikartojimo dažnių italų kalbos įrašuose pasiskirstymą (žr. 3.27 pav.) išsiaiškinta, kad komponentės pirmojo kvartilio reikšmė – tik 0,708%, antrojo – 1,173%, trečiojo – 1,686%.

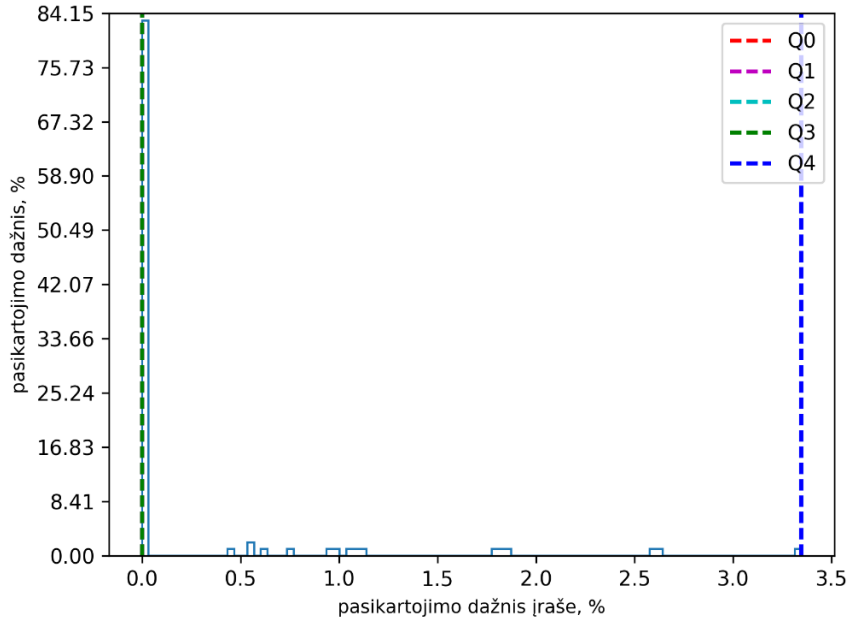
Dažniausiai pasitaikančios italų kalbos moterų garso įrašų balso sričių įvairių akustinių modelių komponentės ir jų užimamos dalys pateikiamos 3.5 lentelėje.

3.5 lentelė. Italų kalbos moterų įrašuose dažniausiai pasitaikančios balso sričių komponentės

Modelis	Balso sričių komponentės	Užima moterų įrašų balso sričių, %
1	4, 84, 603, 1845, 1980, 1027	8,9
2	829, 1012, 311, 865, 1360, 1275	7
5	0	100
6	924, 759, 908, 139, 1173, 293	8,9

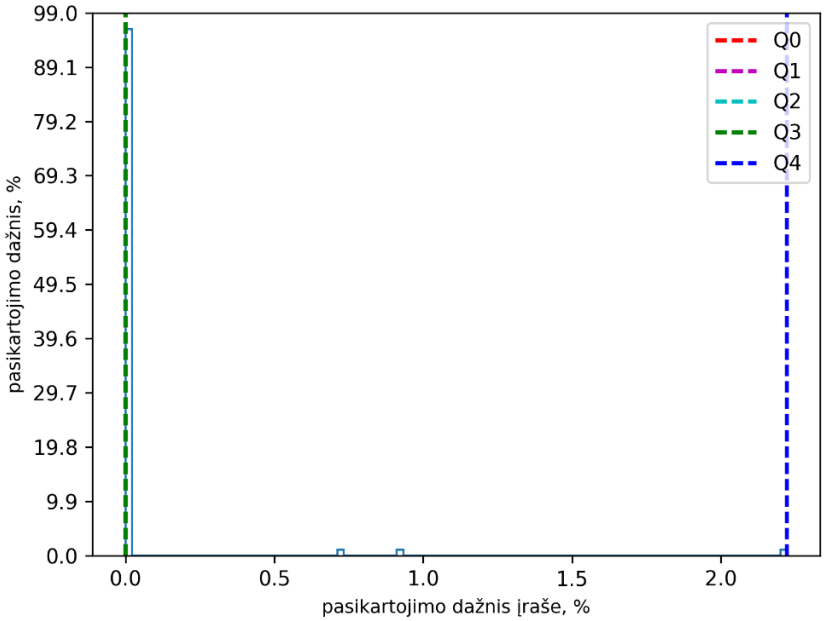
Pagal lentelės rezultatus nustatyta, kad 5-ojo modelio dažniausiai pasitaikančios balso sričių komponentės nenustatytos, kadangi italų kalbos garso įrašuose mažai pašalinių triukšmų. Dėl to dažniausiai pasitaikančių italų kalbos įrašų komponenčių lyginamoji analizė atliekama be šio modelio. Taigi nustatyta, kad dažniausiai pasitaikančios moterų balso sričių 2-ojo modelio komponentės užima mažiausiai moterų įrašų balso sričių – 7%, daugiausiai užima 1-ojo ir 6-ojo modelio komponentės – po 8,9%.

Moterų fono sričių 1-ojo modelio komponenčių 2046, 685 ir 682 pasikartojimo dažnių įrašuose pasiskirstymai pateikiami 3.28, 3.29 ir 3.30 paveiksluose.



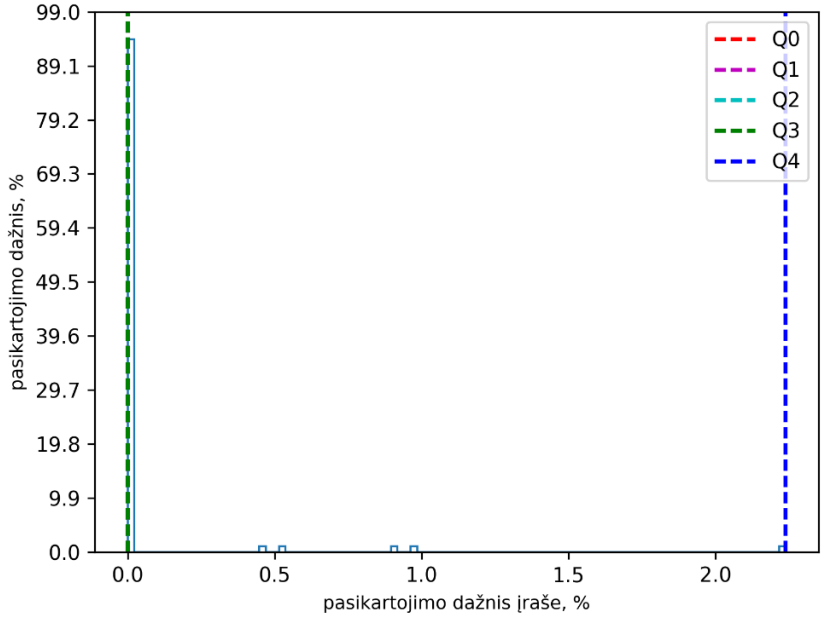
3.28 pav. Moterų fono srities 1-ojo modelio komponentės 2046 pasikartojimo dažnių įrašuose pasiskirstymas

Išnagrinėjus moterų fono srities 1-ojo modelio komponentės 2046 pasikartojimo dažnių įrašuose pasiskirstymą (žr. 3.28 pav.) išsiaiškinta, kad komponentės pirmojo, antrojo ir trečiojo kvartilių reikšmės tokios pat (0%).



3.29 pav. Moterų fono srities 1-ojo modelio komponentės 685 pasikartojimo dažnių įrašuose pasiskirstymas

Ištyrus moterų fono srities 1-ojo modelio komponentės 685 pasikartojimo dažnių įrašuose pasiskirstymą (žr. 3.29 pav.) nustatyta, kad komponentės pasikartojimo dažnių įrašuose pirmojo, antrojo ir trečiojo kvartilių reikšmės – 0%.



3.30 pav. Moterų fono srities 1-ojo modelio komponentės 682 pasikartojimo dažnių įrašuose pasiskirstymas

Analizuojant moterų fono srities 1-ojo modelio komponentės 682 pasikartojimų dažnių įrašuose pasiskirstymą (žr. 3.30 pav.) pastebėta, kad komponentės pirmojo, antrojo ir trečiojo kvartilių reikšmės vėlgi tik 0%.

Atlikus tyrimą nustatytos italų kalbos moterų garso įrašų fono srityse dominuojančios fono sričių įvairių akustinių modelių komponentės (žr. 3.6 lentelė).

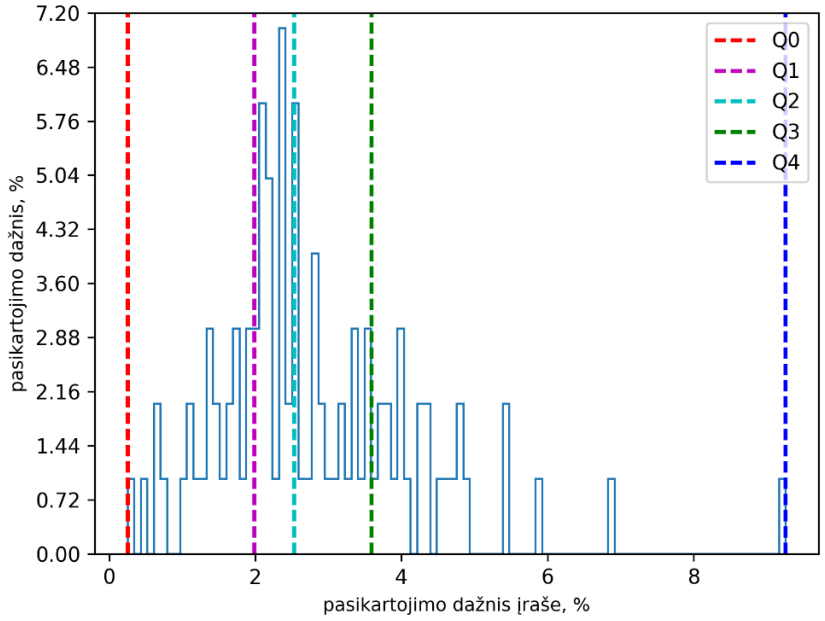
3.6 lentelė. Italų kalbos moterų įrašuose dažniausiai pasitaikančios fono sričių komponentės

Modelis	Fono sričių komponentės	Užima moterų įrašų fono sričių, %
1	1938, 853, 1386, 1363, 1629, 1235	13,8
2	1766, 912, 1218, 2017, 1687, 201	17,4
5	0	100
6	730, 1597, 1673, 475, 684, 26	17,4

Išsiaiškinta, jog dažniausiai pasitaikančios moterų fono sričių 2-ojo ir 6-ojo modelio komponentės užima daugiausiai moterų įrašų fono sričių (po 17,4%), o mažiausiai užima 1-ojo modelio komponentės (13,8%).

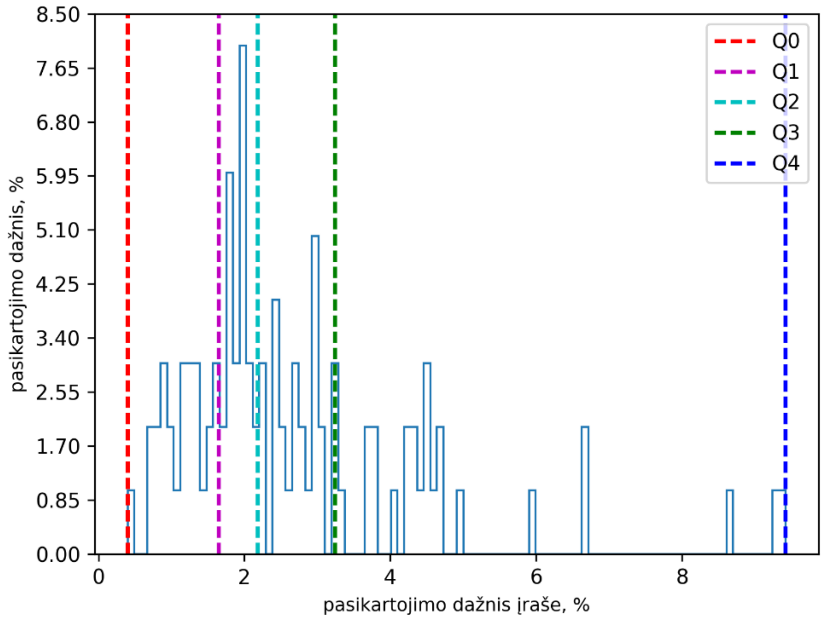
Moterų įrašų balso sričių komponentės skiriasi nuo fono sričių komponenčių. Belieka įsitikinti, ar ir vyrų įrašų balso sričių komponentės skiriasi nuo fono sričių komponenčių.

Žemiau paveiksluose pateikiami vyrų balso sričių 1-ojo modelio komponenčių 1060, 548 ir 676 pasikartojimo dažnių italų kalbos įrašuose pasiskirstymai.



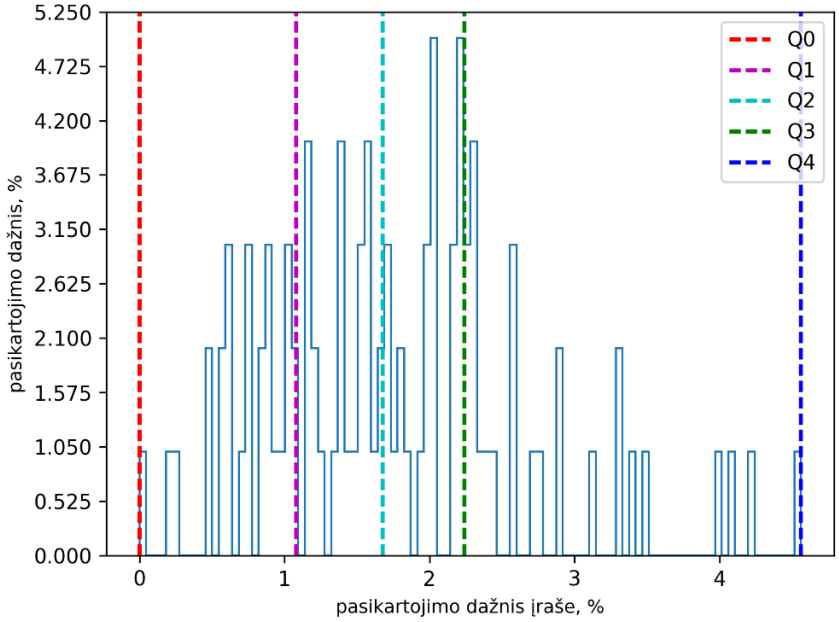
3.31 pav. Vyrų balso srities 1-ojo modelio komponentės 1060 pasikartojimo dažnių įrašuose pasiskirstymas

Ištyrus vyrų balso srities 1-ojo modelio komponentės 1060 pasikartojimo dažnių įrašuose pasiskirstymą (žr. 3.31 pav.) nustatyta, kad komponentės pirmojo kvartilio reikšmė siekia 1,987%, antrojo kvartilio reikšmė – 2,533%, trečiojo kvartilio – 3,88%.



3.32 pav. Vyrų balso srities 1-ojo modelio komponentės 548 pasikartojimo dažnių įrašuose pasiskirstymas

Analizuojant vyrų balso srities 1-ojo modelio komponentės 548 pasikartojimo dažnių įrašuose pasiskirstymą (žr. 3.32 pav.) nustatyta, kad komponentės pirmojo kvartilio reikšmė – tik 1,652%, antrojo kvartilio – 2,183%, trečiojo kvartilio – 3,458%.



3.33 pav. Vyrų balso srities 1-ojo modelio komponentės 676 pasikartojimo dažnių įrašuose pasiskirstymas

Išnagrinėjus vyrų balso srities 1-ojo modelio komponentės 676 pasikartojimo dažnių įrašuose pasiskirstymą (žr. 3.33 pav.) nustatyta, kad komponentės pirmojo kvartilio reikšmė – 1,082%, antrojo kvartilio – 1,676%, trečiojo – 2,409%.

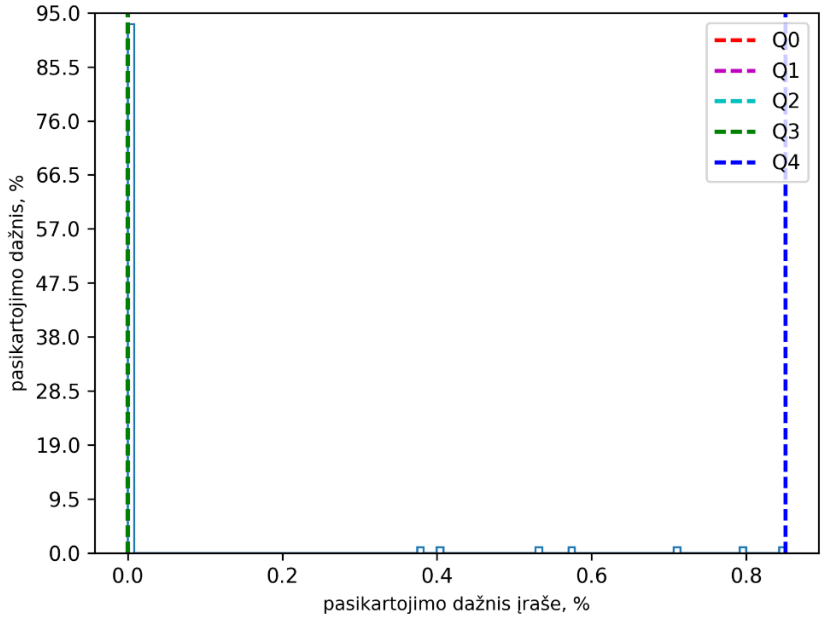
Nustatytos vyrų įrašų balso srityse dominuojančios įvairių akustinių modelių komponentės ir jų užimamos dalys (žr. 3.7 lentelė).

3.7 lentelė. Italų kalbos vyrų įrašuose dažniausiai pasitaikančios balso sričių komponentės

Modelis	Balso sričių komponentės	Užima vyrų įrašų balso sričių, %
1	676, 164, 1420, 548, 1311, 1060	11,1
2	707, 915, 1850, 1344, 65, 812	4,5
5	0	100
6	77, 1992, 569, 488, 1256, 1401	5,9

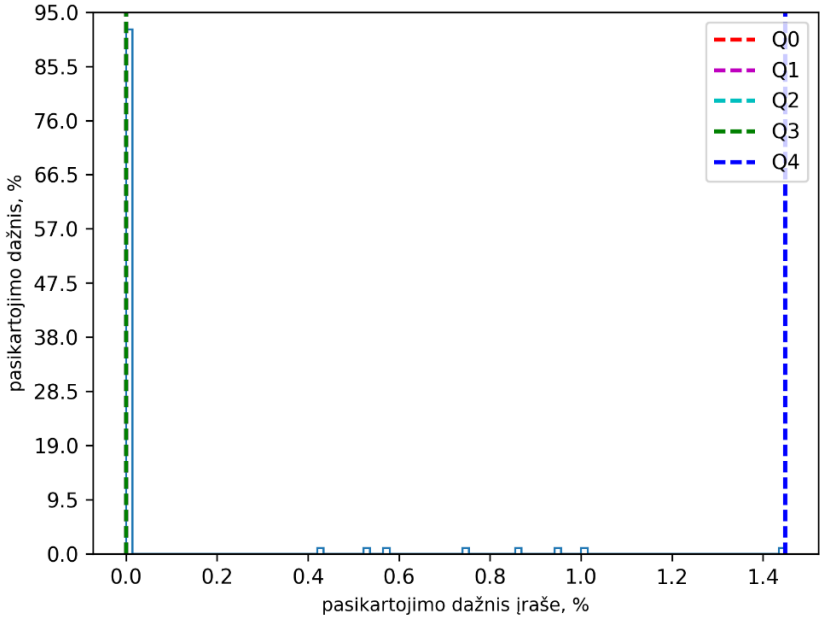
Išsiaiškinta, kad dažniausiai pasitaikančios 6 vyrų balso sričių 2-ojo modelio komponentės užima mažiausiai vyrų įrašų balso sričių (4,5%), o daugiausiai vyrų balso sričių užima 1-ojo modelio komponentės (11,1%).

Vyrų fono sričių 1-ojo modelio komponenčių 2042, 642 ir 690 pasikartojimo dažnių įrašuose pasiskirstymai vaizduojami 3.34, 3.35 ir 3.36 paveiksluose.



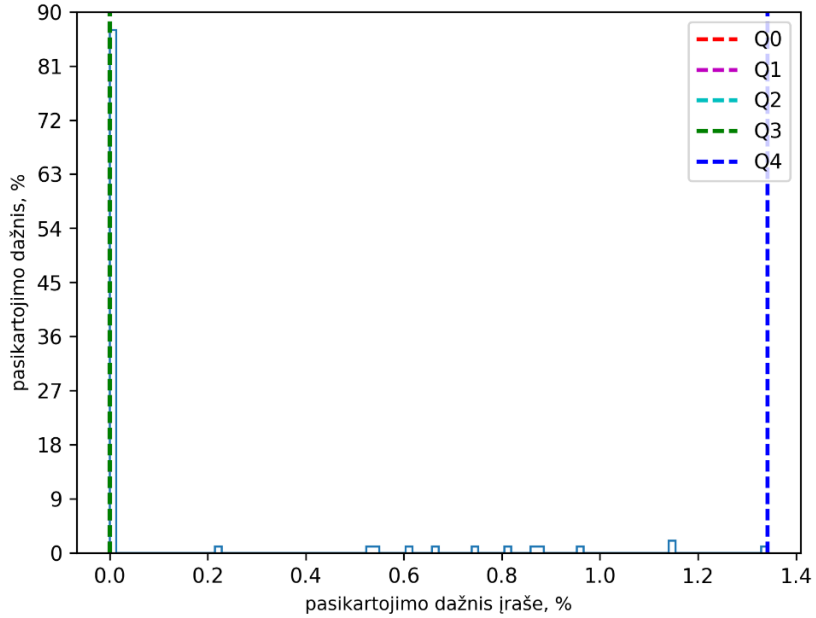
3.34 pav. Vyrų fono srities 1-ojo modelio komponentės 2042 pasikartojimo dažnių įrašuose pasiskirstymas

Tiriant 3.34 paveikslą išsiaiškinta, kad vyrų fono srities 1-ojo modelio komponentė 2042 pasitaiko retai, kadangi komponentės pasikartojimo dažnių įrašuose pirmojo, antrojo ir trečiojo kvartilių reikšmės – 0%.



3.35 pav. Vyrų fono srities 1-ojo modelio komponentės 642 pasikartojimo dažnių įrašuose pasiskirstymas

Išanalizavus 3.35 paveikslą nustatyta, kad vyrų fono srities 1-ojo modelio komponentės 642 pasikartojimo dažnių įrašuose pirmojo, antrojo ir trečiojo kvartilių reikšmės – 0%.



3.36 pav. Vyrų fono srities 1-ojo modelio komponentės 690 pasikartojimo dažnių įrašuose pasiskirstymas

Vyrų fono srities 1-ojo modelio komponentės 690 pasikartojimo dažnių įrašuose pirmojo, antrojo ir trečiojo kvartilių reikšmės tokios pat (0%).

Vyrų įrašų fono srityse dominuojančios komponentės pateikiamos 3.8 lentelėje.

3.8 lentelė. Italų kalbos vyrų įrašuose dažniausiai pasitaikančios fono sričių komponentės

Modelis	Fono sričių komponentės	Užima vyrų įrašų fono sričių, %
1	837, 1611, 1057, 1341, 651, 722	13,8
2	243, 560, 1497, 1542, 671, 1363	14,5
5	0	100
6	1001, 1992, 1291, 77, 1121, 1128	22,5

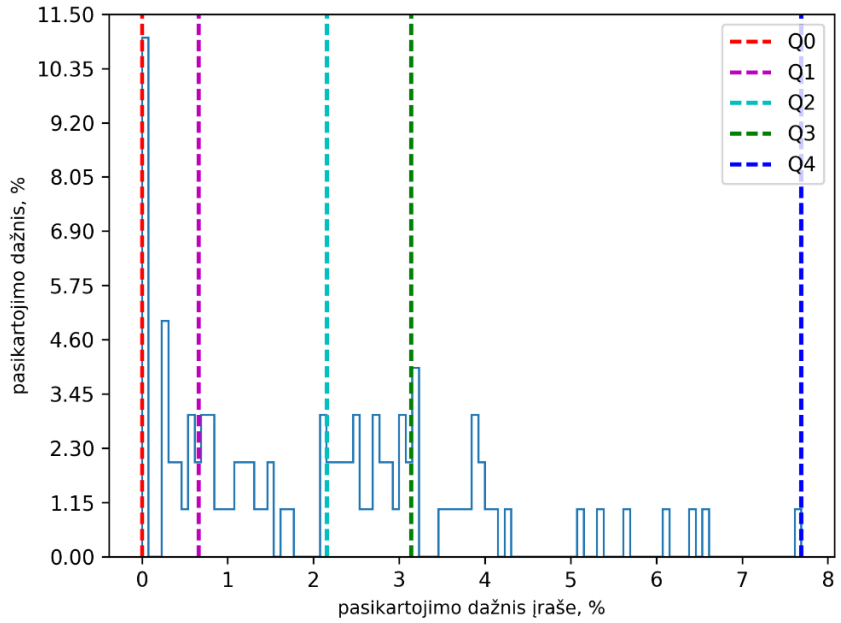
Pagal lentelėje pateiktus rezultatus nustatyta, kad dažniausiai pasitaikančios 6 vyrų fono sričių 6-ojo komponentės užima daugiausiai vyrų įrašų fono sričių (22,5%), mažiausiai vyrų įrašų fono sričių užima vyrų fono sričių 1-ojo modelio komponentės (13,8%).

Išsiaiškinta, kad italų kalbos įrašų balso sričių 1-ojo modelio komponentės skiriasi nuo fono sričių 1-ojo modelio komponenčių, o dažniausiai pasitaikančios balso ar fono sričių skirtingų akustinių modelių komponentės užima nevienodas įrašų balso ar fono sričių dalis.

3.1.4. Prancūzų kalbos garso įrašų rezultatai

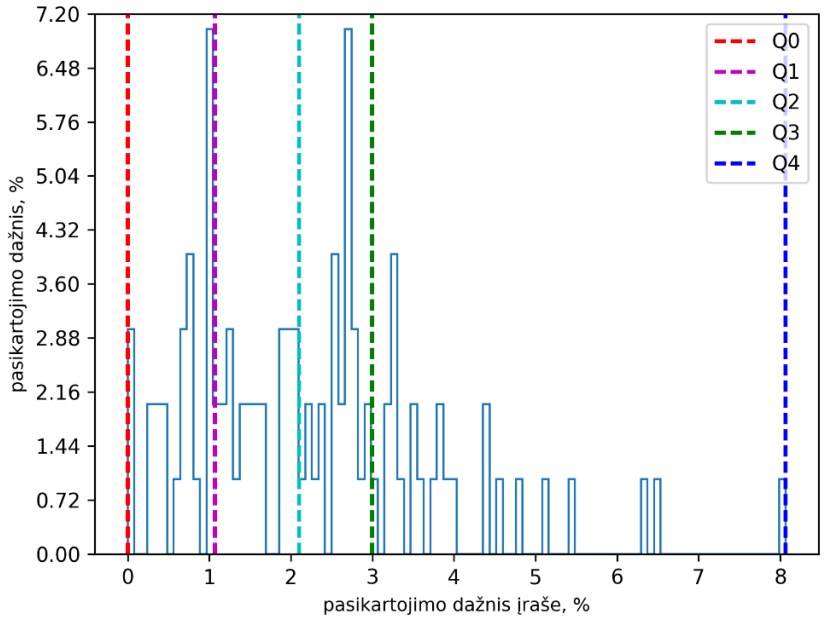
Kad išskirti prancūzų kalba parinktų pokalbių įrašų požymius ir apskaičiuoti akustinio modelio komponenčių statistikas tirta po 100 vyrų ir moterų įrašų. Pirma aptariamos moterų įrašų analizės metu išskirtos komponentės.

Apačioje pateiktuose paveiksluose vaizduojami moterų balso sričių 1-ojo modelio komponenčių 1060, 548 ir 1311 pasikartojimo dažnių įrašuose pasiskirstymai.



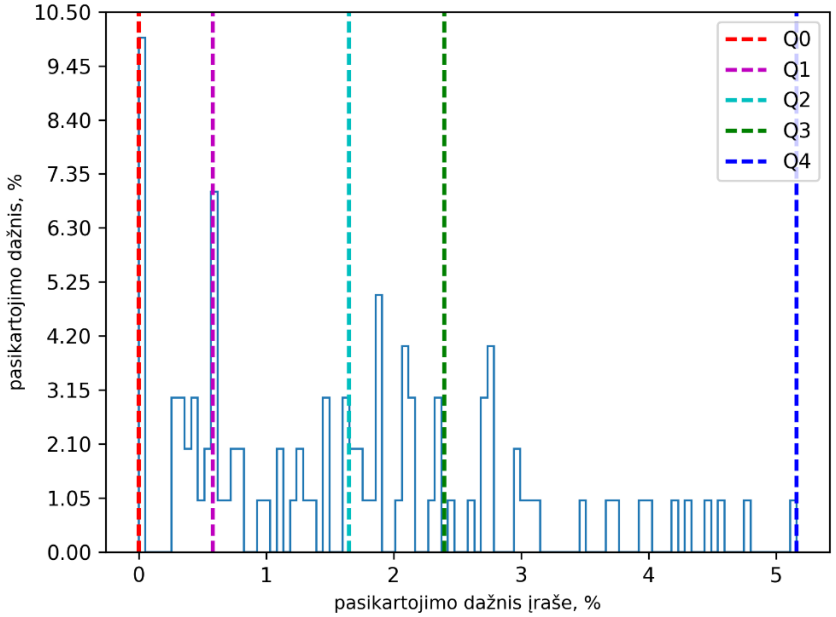
3.37 pav. Moterų balso srities 1-ojo modelio komponentės 1060 pasikartojimo dažnių įrašuose pasiskirstymas

Moterų balso srities 1-ojo modelio komponentės 1060 pasikartojimo dažnių įrašuose pasiskirstymo rezultatai (žr. 3.37 pav.) atskleidė, kad komponentės pasikartojimo dažnių įrašuose pirmojo kvartilio reikšmė – tik 0,664%, antrojo – 2,157%, trečiojo – 3,14%.



3.38 pav. Moterų balso srities 1-ojo modelio komponentės 548 pasikartojimo dažnių įrašuose pasiskirstymas

Išnagrinėjus 3.38 paveikslą nustatyta, kad moterų balso srities 1-ojo modelio komponentės 548 pasikartojimo dažnių įrašuose pirmojo kvartilio reikšmė siekia 1,068%, antrojo kvartilio – 2,1%, trečiojo kvartilio – 2,993%.



3.39 pav. Moterų balso srities 1-ojo modelio komponentės 1311 pasikartojimo dažnių įrašuose pasiskirstymas

Analizuojant balso srities 1-ojo modelio komponentės 1311 pasikartojimo dažnių įrašuose pasiskirstymą (žr. 3.39 pav.) nustatyta, kad komponentės pirmojo kvartilio reikšmė – tik 0,581%, antrojo kvartilio – 1,648%, trečiojo – 2,396%.

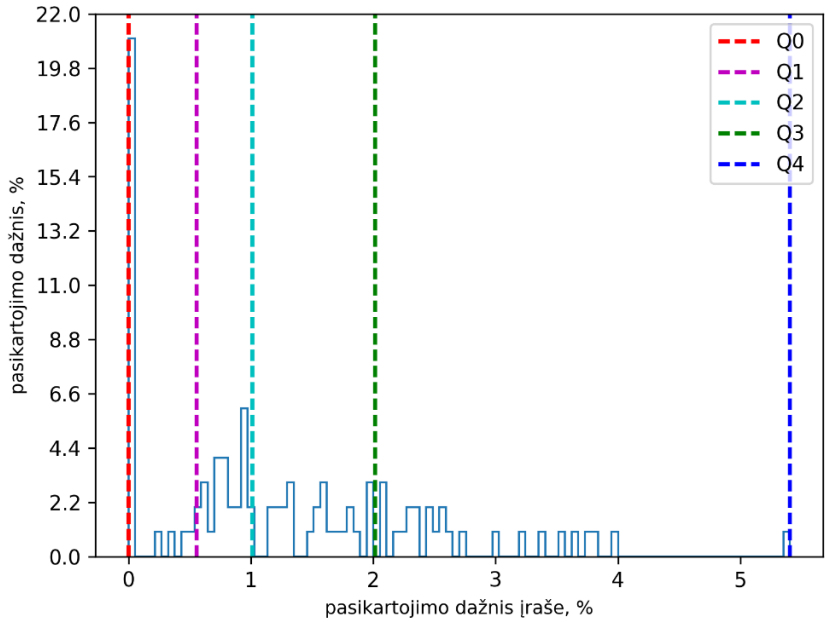
Dažniausiai pasitaikančios moterų balso sričių 1-ojo, 2-ojo, 7-ojo ir 8-ojo modelio komponentės pateikiamos 3.9 lentelėje.

3.9 lentelė. Prancūzų kalbos moterų įrašuose dažniausiai pasitaikančios balso sričių komponentės

Modelis	Balso sričių komponentės	Užima moterų įrašų balso sričių, %
1	1060, 548, 1572, 1311, 1796, 676	19,7
2	1529, 246, 1670, 1335, 1234, 1031	4,6
7	0	100
8	1363, 1325, 328, 823, 550, 650	7,7

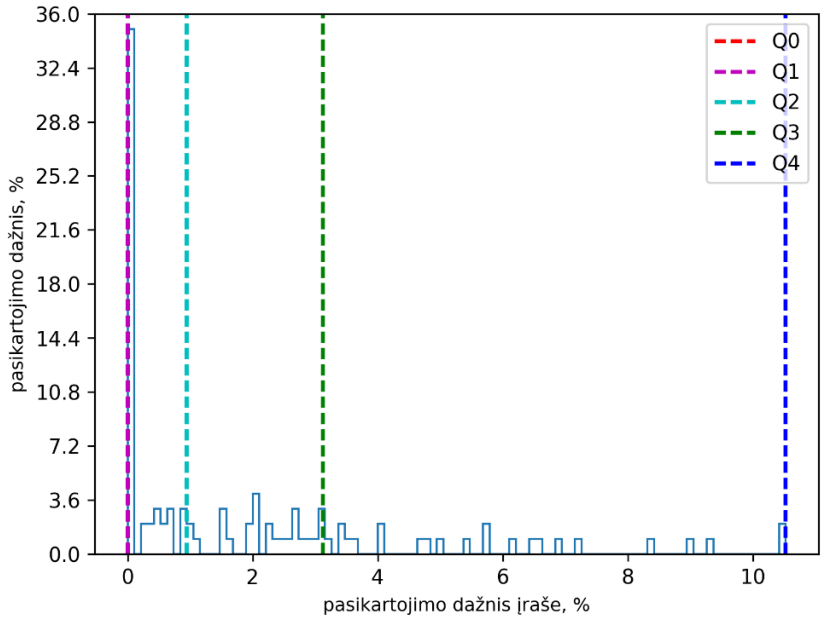
Išanalizavus lentelės duomenis išsiaiškinta, jog 7-ojo modelio vyraujančios komponentės nenustatytos, kadangi parinktuose įrašuose nėra pašalinių triukšmų, todėl toliau analizė atliekama be šio modelio rezultatų. Buvo nustatyta, kad dažniausiai pasitaikančios moterų balso sričių 2-ojo modelio komponentės užima mažiausiai moterų įrašų balso sričių (4,6%), o daugiausiai moterų įrašų balso sričių užima 1-ojo modelio komponentės (19,7%).

Moterų fono sričių 1-ojo modelio komponenčių 1934, 1662 ir 798 pasikartojimo dažnių prancūzų kalbos įrašuose pasiskirstymai pateikiami 3.40, 3.41 ir 3.42 paveiksluose.



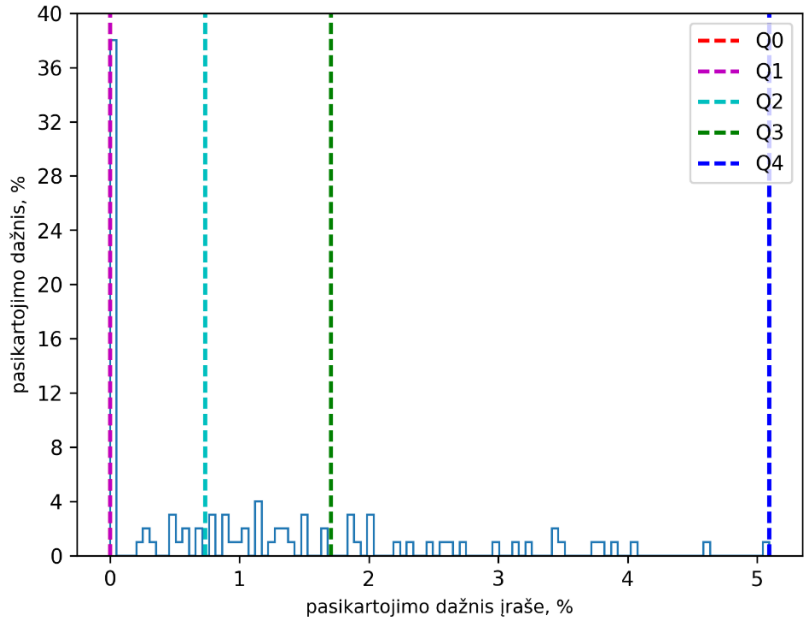
3.40 pav. Moterų fono srities 1-ojo modelio komponentės 1934 pasikartojimo dažnių įrašuose pasiskirstymas

Pastebėta, kad moterų fono srities 1-ojo modelio komponentės 1934 pasikartojimo dažnių įrašuose pirmojo kvartilio reikšmė – tik 0,555%, antrojo – 1,012%, trečiojo – 2,014%.



3.41 pav. Moterų fono srities 1-ojo modelio komponentės 1662 pasikartojimo dažnių įrašuose pasiskirstymas

Ištyrus moterų fono srities 1-ojo modelio komponentės 1662 pasikartojimo dažnių įrašuose pasiskirstymą (žr. 3.41 pav.) nustatyta, kad komponentės pasikartojimo dažnių įrašuose pirmojo kvartilio reikšmė 0%, antrojo kvartilio – tik 0,945%, trečiojo – 3,125%.



3.42 pav. Moterų fono srities 1-ojo modelio komponentės 798 pasikartojimo dažnių įrašuose pasiskirstymas

Moterų fono srities 1-ojo modelio komponentės 798 pasikartojimo dažnių įrašuose pirmojo kvartilio reikšmė – 0%, antrojo – 0,737%, trečiojo – 1,709%.

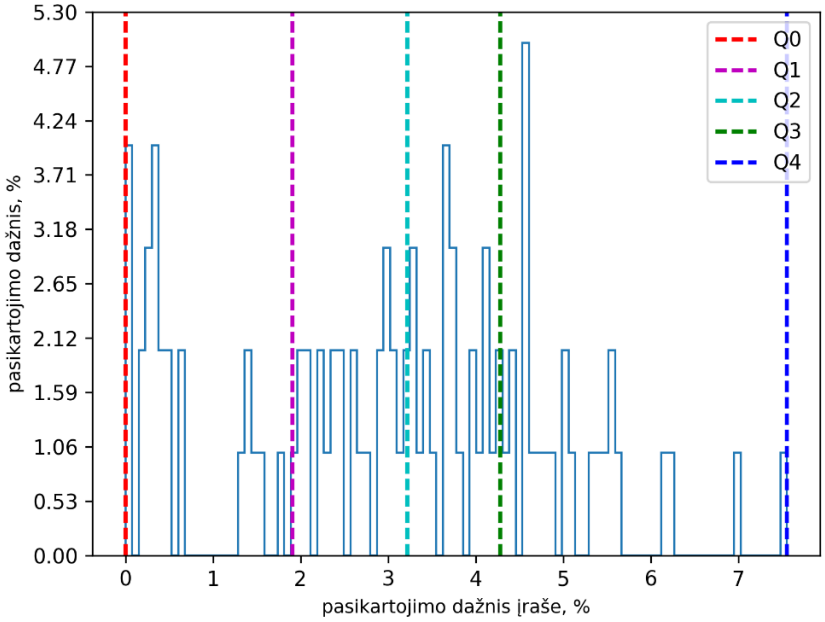
Nustatytos moterų fono srityse dažniausiai pasitaikančios įvairių akustinių modelių komponentės (žr. 3.10 lentelė).

3.10 lentelė. Prancūzų kalbos moterų įrašuose dažniausiai pasitaikančios fono sričių komponentės

Modelis	Fono sričių komponentės	Užima moterų įrašų fono sričių, %
1	2046, 334, 1854, 2014, 1997, 798	26,1
2	606, 852, 579, 1501, 1283, 1204	17,1
7	0	100
8	1818, 1111, 1763, 45, 1949, 1648	27

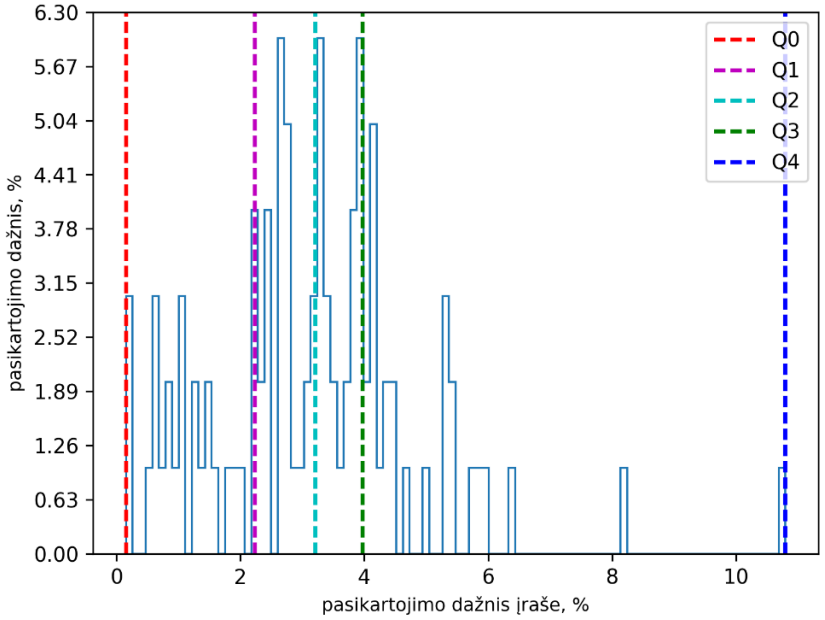
Pagal lentelės duomenis nustatyta, jog dažniausiai pasitaikančios moterų fono sričių 8-ojo modelio komponentės užima daugiausiai moterų įrašų fono sričių – 27%, o mažiausiai moterų įrašų fono sričių užima 2-ojo modelio komponentės – 17,1%.

Toliau aptariami vyrų įrašų tyrimo metu gauti rezultatai, t. y. pateikiami komponenčių pasikartojimo dažnių įrašuose pasiskirstymo grafikai medianos atžvilgiu ir nustatomos dažniausiai pasitaikančios balso ir fono sričių komponentės. Apačioje vaizduojami vyrų balso sričių 1-ojo modelio komponenčių 1060, 548 ir 1311 pasikartojimo dažnių įrašuose pasiskirstymai.



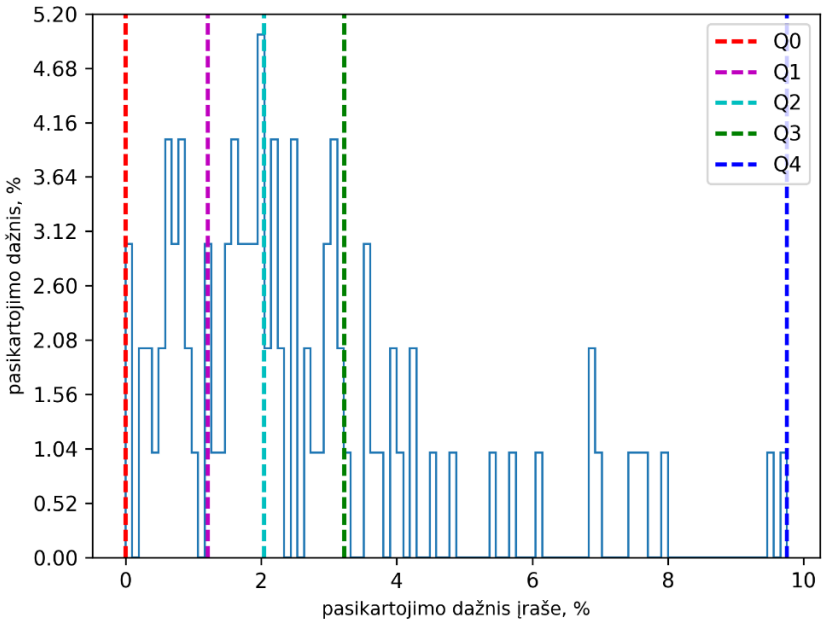
3.43 pav. Vyrų balso srities 1-ojo modelio komponentės 1060 pasikartojimo dažnių įrašuose pasiskirstymas

Išanalizavus vyrų balso srities 1-ojo modelio komponentės 1060 pasikartojimo dažnių prancūzų kalbos įrašuose pasiskirstymą (žr. 3.43 pav.) nustatyta, kad komponentės pirmojo kvartilio reikšmė siekia tik 1,905%, antrojo kvartilio – 3,221%, trečiojo – 4,279%.



3.44 pav. Vyrų balso srities 1-ojo modelio komponentės 548 pasikartojimo dažnių įrašuose pasiskirstymas

Analizuojant 3.44 paveikslą pastebėta, jog vyrų balso srities 1-ojo modelio komponentės 548 pasikartojimo dažnių įrašuose pirmojo kvartilio reikšmė – 2,233%, antrojo – 3,209%, trečiojo – 3,97%.



3.45 pav. Vyrų balso srities 1-ojo modelio komponentės 1311 pasikartojimo dažnių įrašuose pasiskirstymas

Išnagrinėjus vyrų balso srities 1-ojo modelio komponentės 1311 pasikartojimo dažnių prancūzų kalbos įrašuose pasiskirstymą (žr. 3.45 pav.) nustatyta, kad komponentės pirmojo kvartilio reikšmė siekia 1,212%, antrojo kvartilio – 2,04%, trečiojo – 3,224%.

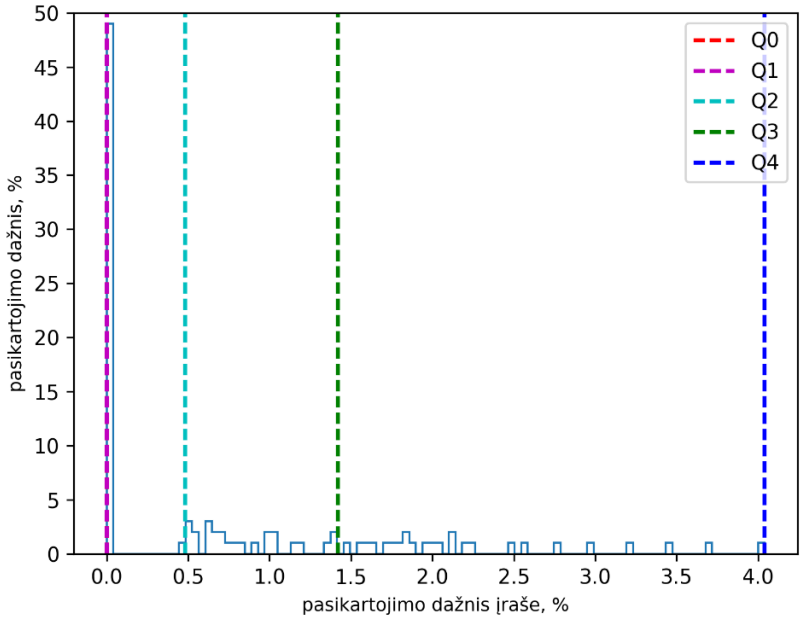
Nustatytos prancūziškai kalbančių vyrų įrašų balso srityse dominuojančios įvairių akustinių modelių komponentės (žr. 3.11 lentelė).

3.11 lentelė. Prancūzų kalbos vyrų įrašuose dažniausiai pasitaikančios balso sričių komponentės

Modelis	Balso sričių komponentės	Užima vyrų įrašų balso sričių, %
1	548, 1060, 1311, 36, 676, 151	18,4
2	661, 832, 363, 1988, 1978, 763	6,1
7	0	100
8	1595, 800, 35, 25, 1734, 134	7,9

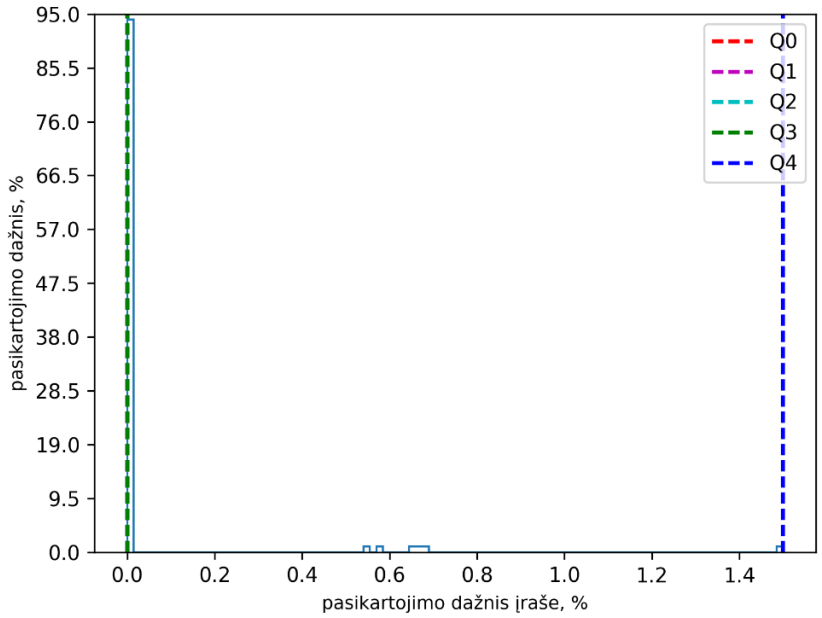
Išanalizavus lentelės duomenis išsiaiškinta, kad dažniausiai pasitaikančios vyrų balso sričių 2-ojo modelio komponentės užima mažiausiai vyrų įrašų balso sričių – 6,1%, daugiausiai vyrų įrašų balso sričių užima 1-ojo modelio komponentės – 18,4%.

Vyrų fono sričių 1-ojo modelio komponenčių 656, 2045 ir 665 pasikartojimo dažnių įrašuose pasiskirstymai vaizduojami 3.46, 3.47 ir 3.48 paveiksluose.



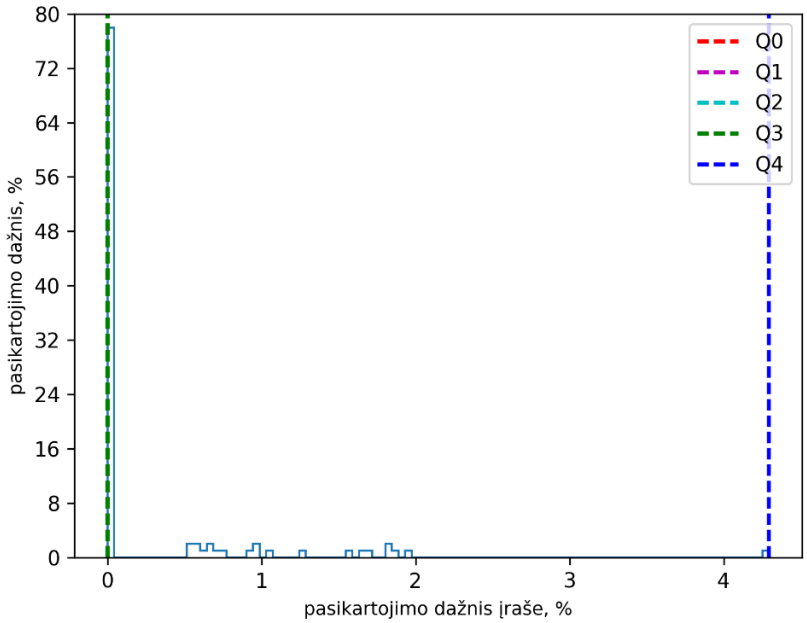
3.46 pav. Vyrų fono srities 1-ojo modelio komponentės 656 pasikartojimo dažnių įrašuose pasiskirstymas

Tiriant vyrų fono srities 1-ojo modelio komponentės 656 pasikartojimo dažnių prancūzų kalbos įrašuose pasiskirstymą (žr. 3.46 pav.) išsiaiškinta, kad komponentės pirmojo kvartilio reikšmė – tik 0%, antrojo – 0,482%, trečiojo – 1,42%.



3.47 pav. Vyrų fono srities 1-ojo modelio komponentės 2045 pasikartojimo dažnių įrašuose pasiskirstymas

Išanalizavus 3.47 paveikslą išsiaiškinta, kad vyrų fono srities 1-ojo modelio komponentės 2045 pasikartojimo dažnių įrašuose pirmojo, antrojo ir trečiojo kvartilių reikšmės tokios pat – 0%.



3.48 pav. Vyrų fono srities 1-ojo modelio komponentės 665 pasikartojimo dažnių įrašuose pasiskirstymas

Vyrų fono srities 1-ojo modelio komponentės 665 pasikartojimo dažnių įrašuose (žr. 3.48 pav.) pirmojo, antrojo ir trečiojo kvartilių reikšmės vėlgi 0%.

Dažniausiai pasitaikančios vyrų fono sričių komponentės pateikiamos 3.12 lentelėje.

3.12 lentelė. Prancūzų kalbos vyrų įrašuose dažniausiai pasitaikančios fono sričių komponentės

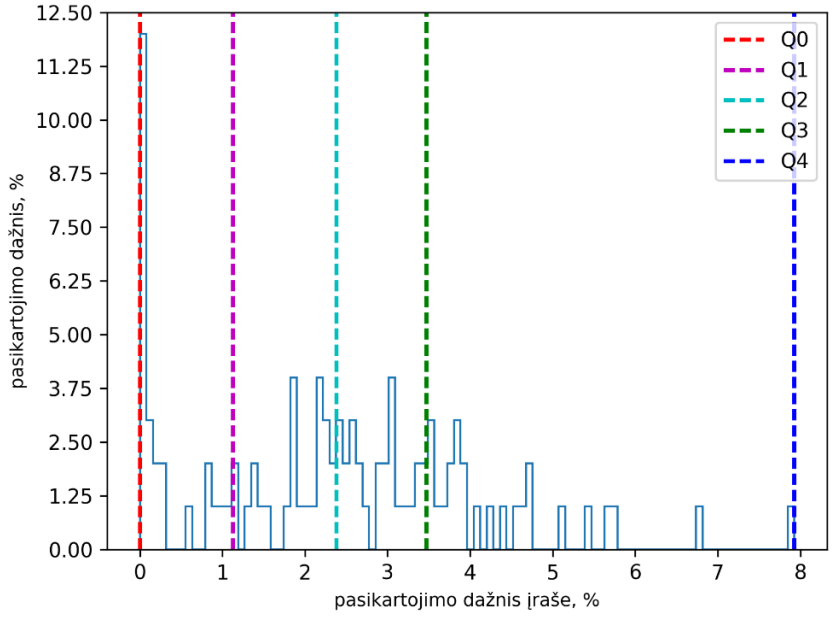
Modelis	Fono sričių komponentės	Užima vyrų įrašų fono sričių, %
1	709, 383, 801, 964, 653, 1057	18,1
2	1840, 171, 1581, 561, 748, 1893	22,9
7	0	100
8	1708, 1035, 1243, 35, 872, 1222	34,9

Pagal lentelės rezultatus nustatyta, kad dažniausiai pasitaikančios vyrų fono sričių 8-ojo modelio komponentės užima didžiausią vyrų įrašų fono sričių dalį (34,9%). Mažiausią fono sričių dalį užima 1-ojo modelio komponentės (18,1%).

Išanalizavus prancūzų kalba parinktus garso įrašus išsiaiškinta, kad dažniausiai pasitaikančios moterų balso sričių 1-ojo modelio komponentės skiriasi nuo fono sričių 1-ojo modelio komponenčių, o dažniausiai pasitaikančios įvairių akustinių modelių komponentės pasiskirsto nevienodai.

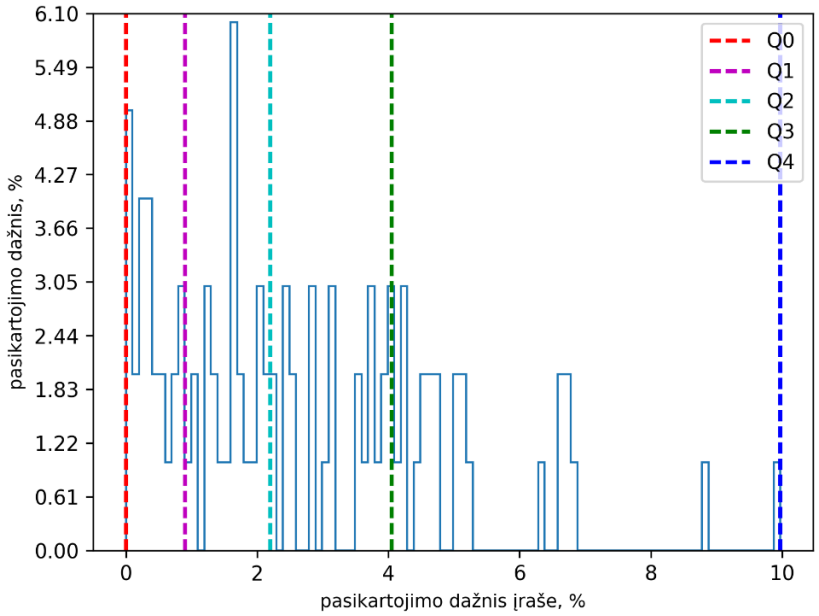
3.1.5. Rusų kalbos garso įrašų rezultatai

Rusų kalbos garso įrašų požymių išskyrimui parinkta po 100 vyrų ir 100 moterų garso įrašų. Komponenčių statistikų tyrimo metu gauti moterų balso sričių 1-ojo modelio komponenčių 1060, 548 ir 676 pasikartojimo dažnių įrašuose pasiskirstymai vaizduojami 3.49, 3.50 ir 3.51 paveiksluose.



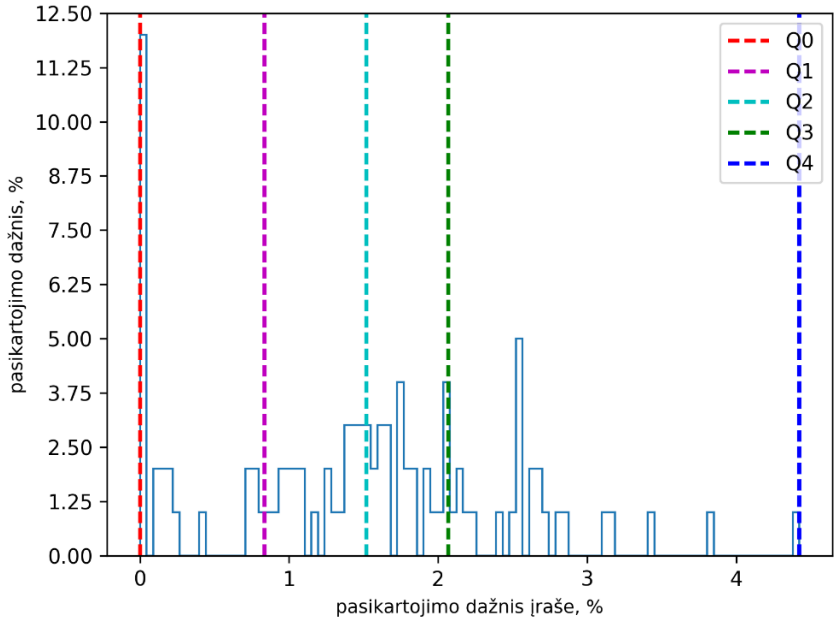
3.49 pav. Moterų balso srities 1-ojo modelio komponentės 1060 pasikartojimo dažnių įrašuose pasiskirstymas

Moterų balso srities 1-ojo modelio komponentės 1060 pasikartojimo dažnių rusų kalbos garso įrašuose pirmojo kvartilio reikšmė – 1,127%, antrojo – 2,381%, trečiojo – 3,473%.



3.50 pav. Moterų balso srities 1-ojo modelio komponentės 548 pasikartojimo dažnių įrašuose pasiskirstymas

Išnagrinėjus 3.50 paveikslą nustatyta, kad moterų balso srities 1-ojo modelio komponentės 548 pasikartojimo dažnių įrašuose pirmojo kvartilio reikšmė tik 0,902%, antrojo kvartilio reikšmė – 2,201%, trečiojo kvartilio – 4,048%.



3.51 pav. Moterų balso srities 1-ojo modelio komponentės 676 pasikartojimo dažnių įrašuose pasiskirstymas

Moterų balso srities 1-ojo modelio komponentės 676 pasikartojimo dažnių įrašuose pirmojo kvartilio reikšmė siekia vos 0,836%, antrojo kvartilio – 1,519%, trečiojo kvartilio – 2,069%.

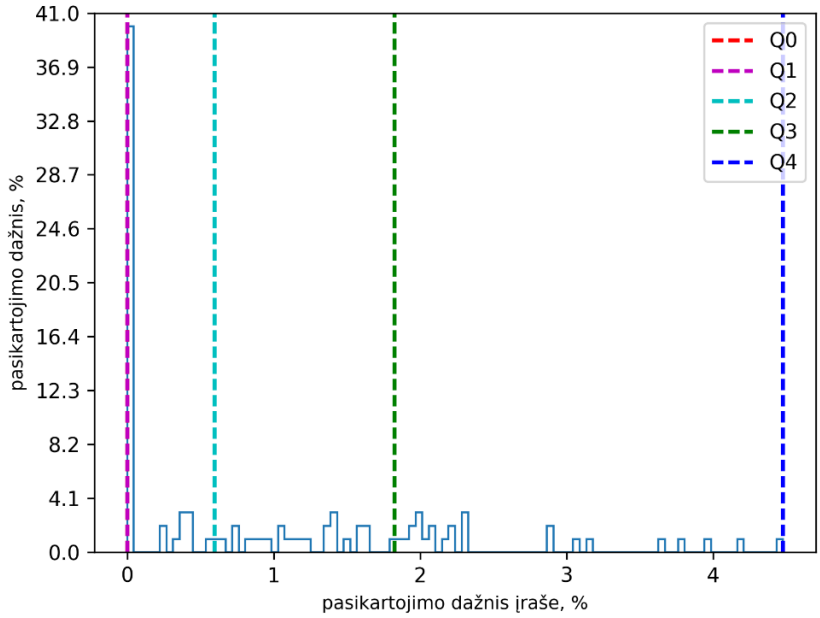
Nustatytos 6 dažniausiai pasitaikančios moterų balso sričių įvairių akustinių modelių komponentės ir apskaičiuotos jų užimamos dalys (žr. 3.13 lentelė).

3.13 lentelė. Rusų kalbos moterų įrašuose dažniausiai pasitaikančios balso sričių komponentės

Modelis	Balso sričių komponentės	Užima moterų įrašų balso sričių, %
1	548, 1060, 164, 1188, 1444, 953	18
2	1554, 815, 1782, 1939, 143, 1993	9,5
9	24, 11, 23, 247, 311, 94	32,7
10	446, 1919, 1659, 1539, 23, 456	10,2

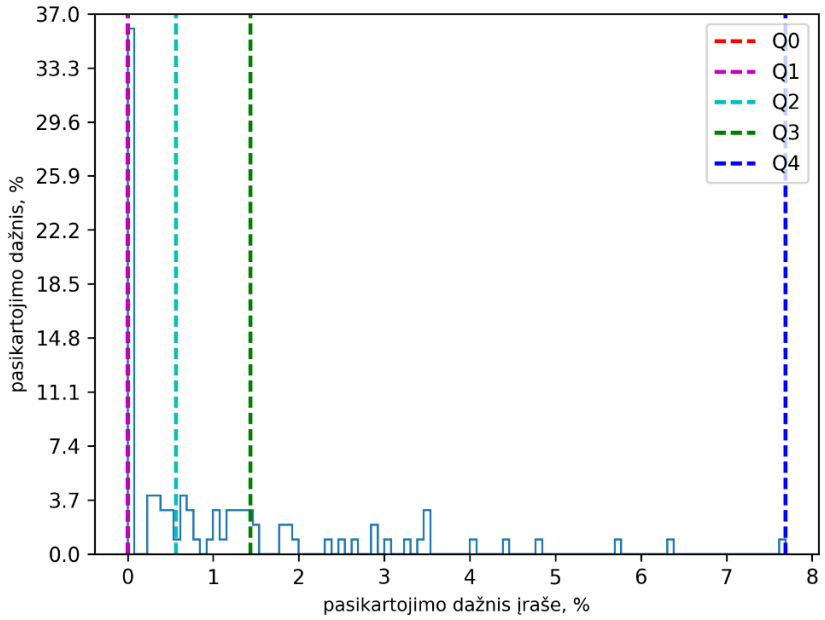
Išsiaiškinta, kad geriausi rezultatai grąžinami 2-uoju modeliu, kadangi dažniausiai pasitaikančios moterų balso sričių 2-ojo modelio komponentės užima mažiausią moterų įrašų balso sričių dalį – 9,5%. Didžiausia moterų įrašų balso sričių dalis užimama 9-ojo modelio komponentėmis – 32,7%.

Moterų fono sričių 1-ojo modelio komponenčių 1060, 1662 bei 1054 pasikartojimo dažnių įrašuose pasiskirstymai pateikiami 3.52, 3.53 ir 3.54 paveiksluose.



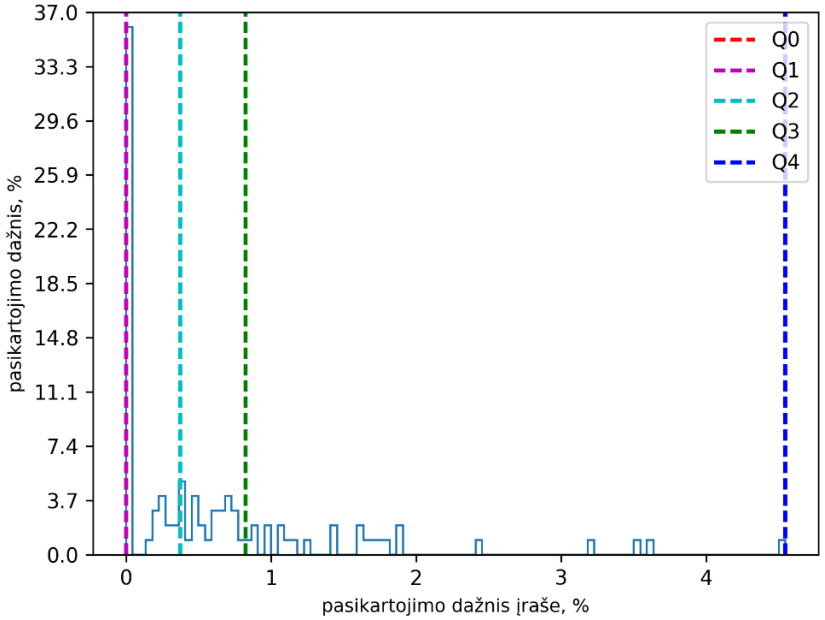
3.52 pav. Moterų fono srities 1-ojo modelio komponentės 1060 pasikartojimo dažnių įrašuose pasiskirstymas

Pagal 3.52 paveikslo rezultatus nustatyta, kad moterų fono srities 1-ojo modelio komponentės 1060 pasikartojimo dažnių rusų kalbos garso įrašuose pirmojo kvartilio reikšmė – 0%, antrojo kvartilio reikšmė siekia tik 0,596%, trečiojo kvartilio reikšmė – 1,824%.



3.53 pav. Moterų fono srities 1-ojo modelio komponentės 1662 pasikartojimo dažnių įrašuose pasiskirstymas

Nustatyta, kad moterų fono srities 1-ojo modelio komponentės 1662 pasikartojimo dažnių įrašuose pirmojo kvartilio reikšmė – 0%, antrojo – 0,567%, trečiojo – 1,435%.



3.54 pav. Moterų fono srities 1-ojo modelio komponentės 1054 pasikartojimo dažnių įrašuose pasiskirstymas

Pagal moterų fono srities 1-ojo modelio komponentės 1054 pasikartojimo dažnių įrašuose pasiskirstymo rezultatus (žr. 3.54 pav.) nustatyta, kad komponentės pirmojo kvartilio reikšmė taip pat 0%, antrojo – 0,375%, trečiojo – 0,824%.

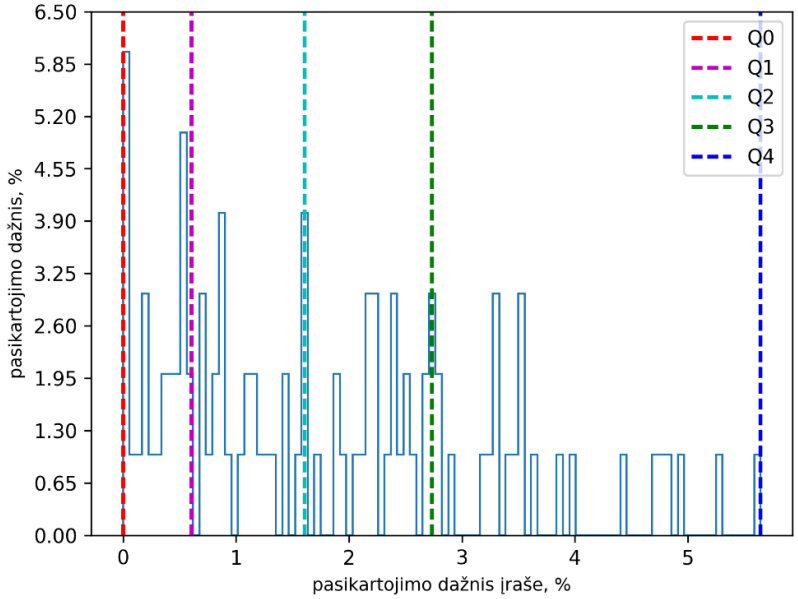
Nustatytos moterų fono srityse dominuojančios 1-ojo, 2-ojo, 9-ojo ir 10-ojo modelio komponentės ir jų užimamų moterų įrašų fono sričių procentinės išraiškos (žr. 3.14 lentelė).

3.14 lentelė. Rusų kalbos moterų įrašuose dažniausiai pasitaikančios fono sričių komponentės

Modelis	Fono sričių komponentės	Užima moterų įrašų fono sričių, %
1	1758, 697, 309, 339, 525, 803	22
2	1923, 1997, 1455, 604, 1821, 1931	19,5
9	604, 860,, 868, 220, 594, 1010	35,8
10	1146, 725, 636, 510, 823, 367	25,2

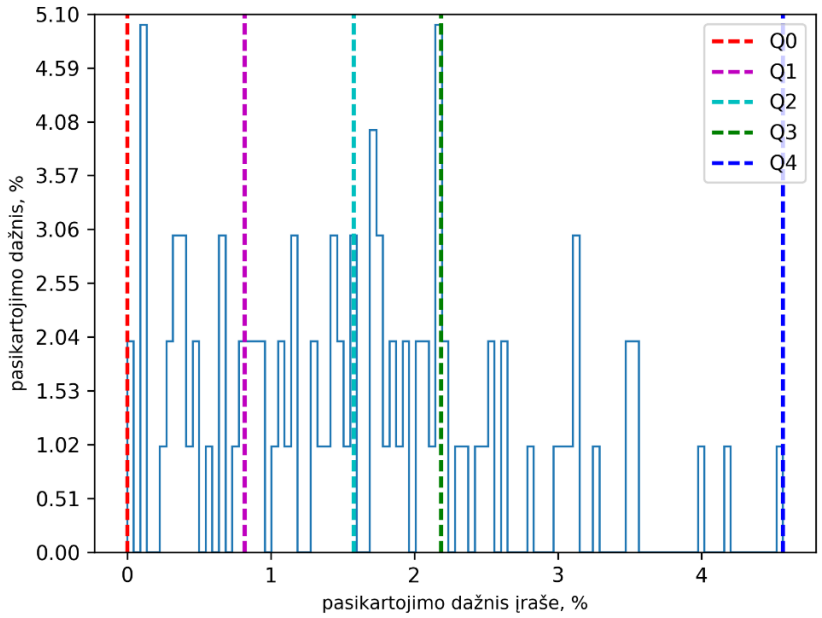
Pagal lentelės duomenis nustatyta, kad dažniausiai pasitaikančios moterų fono sričių 9-ojo modelio komponentės užima daugiausiai moterų įrašų fono sričių – 35,8%, o mažiausiai moterų įrašų fono sričių užima 2-ojo modelio komponentės – 19,5%. Be to išsiaiškinta, kad dažniausiai pasitaikančios moterų balso sričių komponentės skiriasi nuo fono sričių komponenčių.

Pagal vyrų garso įrašų tyrimo metu gautus rezultatus sudaryti balso sričių 1-ojo modelio komponenčių 548 (žr. 3.55), 1060 (žr. 3.56) ir 1420 (žr. 3.57) pasikartojimo dažnių įrašuose pasiskirstymo grafikai.



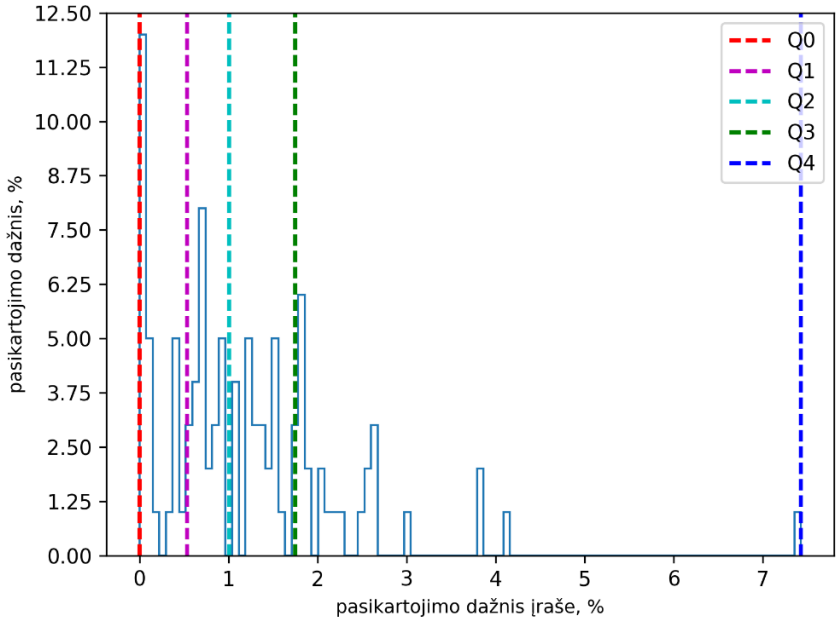
3.55 pav. Vyrų balso srities 1-ojo modelio komponentės 548 pasikartojimo dažnių įrašuose pasiskirstymas

Ištyrus 3.55 paveikslą išsiaiškinta, kad vyrų balso srities 1-ojo modelio komponentės 548 pasikartojimo dažnių rusų kalbos garso įrašuose pirmojo kvartilio reikšmė – tik 0,605%, antrojo kvartilio – 1,607%, trečiojo kvartilio – 2,736%.



3.56 pav. Vyrų balso srities 1-ojo modelio komponentės 1060 pasikartojimo dažnių įrašuose pasiskirstymas

Išanalizavus 3.56 paveikslą pastebėta, jog vyrų balso srities 1-ojo modelio komponentės 1060 pasikartojimo dažnių įrašuose pirmojo kvartilio reikšmė – vos 0,819%, antrojo kvartilio – 1,578%, trečiojo – 2,188%.



3.57 pav. Vyrų balso srities 1-ojo modelio komponentės 1420 pasikartojimo dažnių įrašuose pasiskirstymas

Nustatyta, kad vyrų balso srities 1-ojo modelio komponentės 1420 pasikartojimo dažnių įrašuose pirmojo kvartilio reikšmė 0,535%, antrojo kvartilio – 1,005%, trečiojo kvartilio – tik 1,746%.

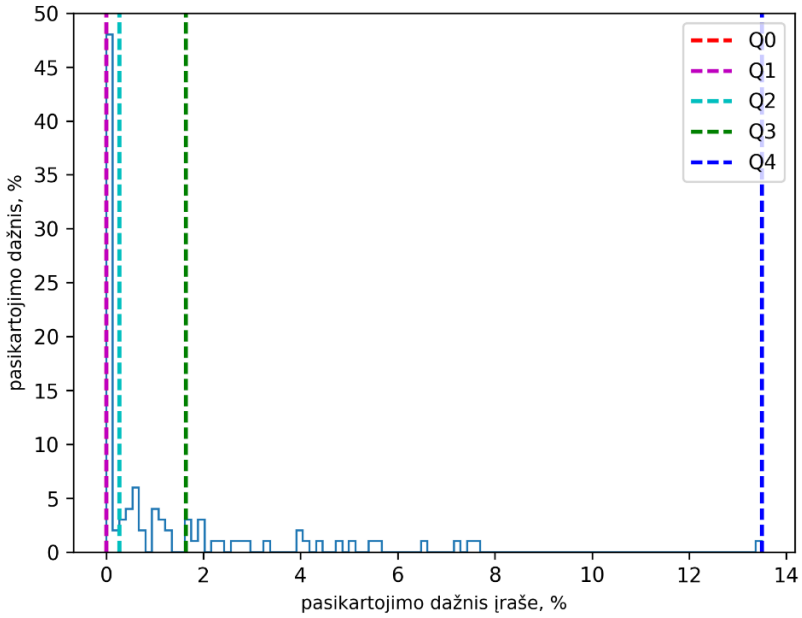
Išsiaiškinta, kokios vyrų balso sričių įvairių akustinių modelių komponentės dažniausiai pasitaiko balso srityse (žr. 3.15 lentelė).

3.15 lentelė. Rusų kalbos vyrų įrašuose dažniausiai pasitaikančios balso sričių komponentės

Modelis	Balso sričių komponentės	Užima vyrų įrašų balso sričių, %
1	1311, 377, 889, 1444, 420, 932	16,6
2	1039, 1299, 1037, 581, 599, 773	5,4
9	84, 111, 119, 559, 574, 311	43,2
10	982, 1034, 1726, 1347, 377, 762	9,9

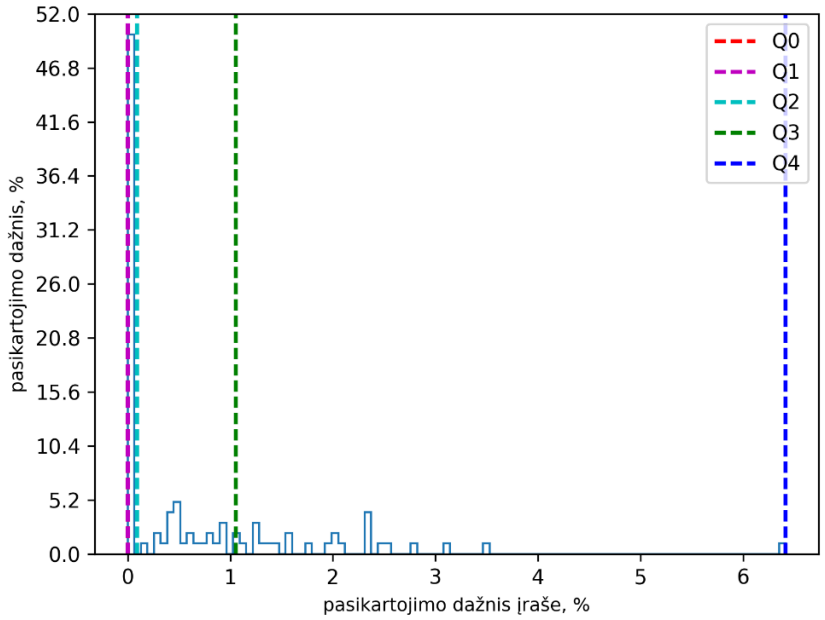
Nustatyta, kad dažniausiai pasitaikančios vyrų balso sričių 2-ojo modelio komponentės užima mažiausią vyrų įrašų balso sričių dalį – 5,4%. Didžiausia vyrų įrašų balso sričių dalis užimama 9-ojo modelio komponentėmis – 43,2%.

Vyrų fono sričių 1-ojo modelio komponenčių (9, 889, 2043) pasikartojimo dažnių įrašuose pasiskirstymai vaizduojami 3.58, 3.59 ir 3.60 paveiksluose.



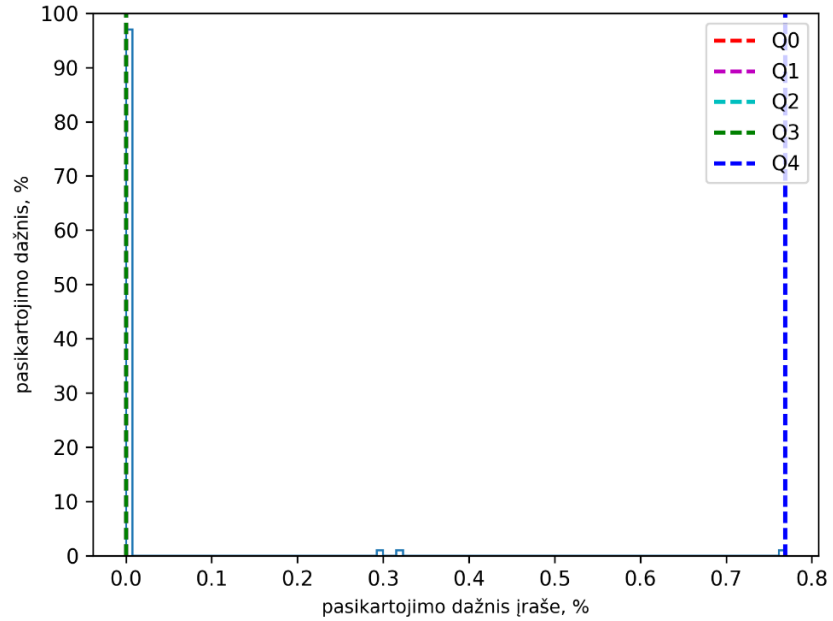
3.58 pav. Vyrų fono srities 1-ojo modelio komponentės 9 pasikartojimo dažnių įrašuose pasiskirstymas

Tiriant vyrų fono srities 1-ojo modelio komponentės 9 pasikartojimo dažnių įrašuose pasiskirstymą (žr. 3.58 pav.) išsiaiškinta, kad komponentės pirmojo kvartilio reikšmė – 0%, antrojo kvartilio reikšmė siekia vos 0,274%, trečiojo kvartilio – 1,643%.



3.59 pav. Vyrų fono srities 1-ojo modelio komponentės 889 pasikartojimo dažnių įrašuose pasiskirstymas

Išanalizavus 3.59 paveiksle vaizduojamą vyrų fono srities 1-ojo modelio komponentės 889 pasikartojimo dažnių įrašuose pasiskirstymą išsiaiškinta, kad komponentės pasikartojimo dažnių įrašuose pirmojo kvartilio reikšmė tik 0%, antrojo – 0,093%, trečiojo – 1,054%.



3.60 pav. Vyrų fono srities 1-ojo modelio komponentės 2043 pasikartojimo dažnių įrašuose pasiskirstymas

Išanalizavus vyrų fono srities 1-ojo modelio komponentės 2043 pasikartojimo dažnių įrašuose pasiskirstymą pastebėta, kad komponentės pirmojo, antrojo ir trečiojo kvartilių reikšmės tokios pat – 0%.

Taikant įvairius modelius nustatytos dominuojančios fono sričių komponentės (žr. 3.16 lentelė).

3.16 lentelė. Rusų kalbos vyrų įrašuose dažniausiai pasitaikančios fono sričių komponentės

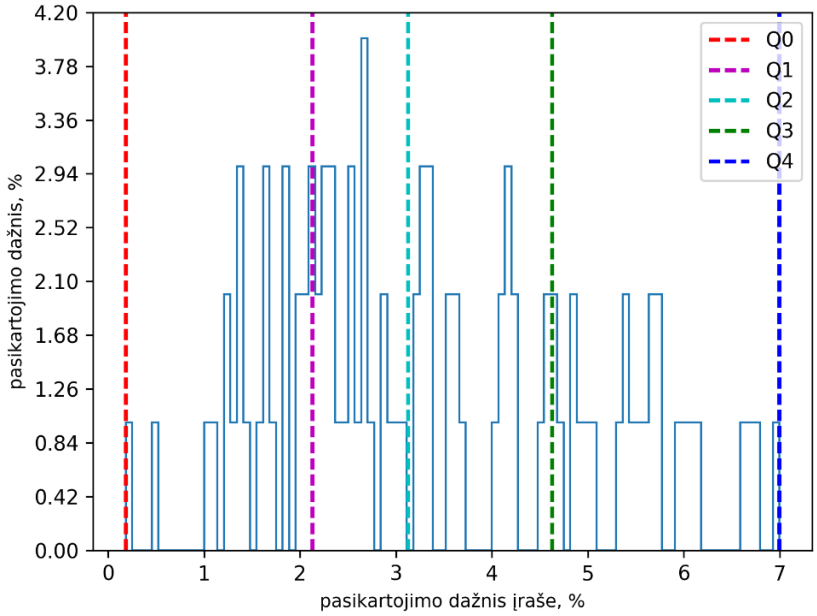
Modelis	Fono sričių komponentės	Užima vyrų įrašų fono sričių, %
1	656, 2041, 1232, 289, 1611, 410	15,2
2	769, 1975, 726, 634, 811, 1358	15,2
9	998, 554, 84, 46, 362, 428	37,6
10	1139, 1583, 1171, 444, 816, 537	23,2

Pagal lentelės rezultatus nustatyta, kad dažniausiai pasitaikančios vyrų fono sričių komponentės užima daugiausiai vyrų garso įrašų fono sričių taikant 9-tąjį modelį – 37,6%, o užima mažiausiai taikant 2-ąjį ir 1-ąjį modelius – 15,2%.

Išnagrinėjus rusų kalba parinktus įrašus nustatyta, kad 1-ojo modelio balso ir fono sričių komponentės skiriasi. Kalbos požymiai po įvairių akustinių modelių komponentes pasiskirsto nevienodai.

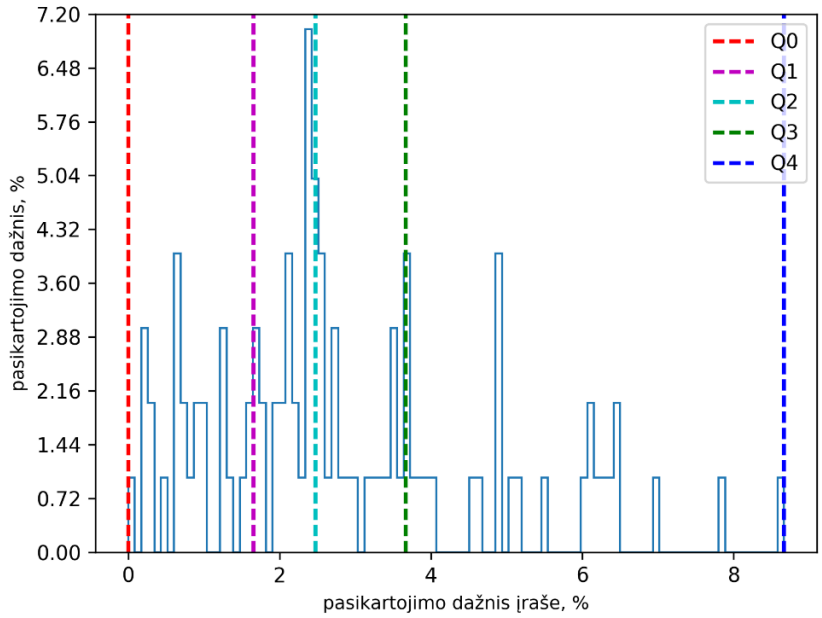
3.1.6. Vokiečių kalbos garso įrašų rezultatai

Požymių išskyrimui atliktas vokiečių kalba parinktų garso įrašų tyrimas. Iš viso tirta po 100 vyrų ir moterų garso įrašų. Moterų balso sričių 1-ojo modelio komponenčių 548, 1060 ir 676 pasikartojimo dažnių įrašuose pasiskirstymai pateikiami 3.61, 3.62 ir 3.63 paveiksluose.



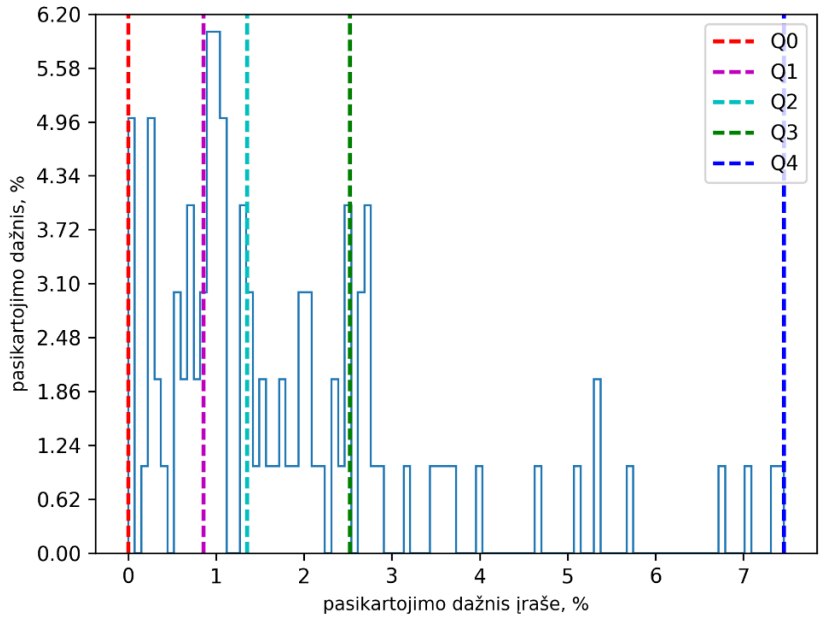
3.61 pav. Moterų balso srities 1-ojo modelio komponentės 548 pasikartojimo dažnių įrašuose pasiskirstymas

Moterų balso srities 1-ojo modelio komponentės 548 pasikartojimo dažnių įrašuose pirmojo kvartilio reikšmė siekia 2,128%, antrojo – 3,13%, trečiojo – 4,626%.



3.62 pav. Moterų balso srities 1-ojo modelio komponentės 1060 pasikartojimo dažnių įrašuose pasiskirstymas

Išnagrinėjus 3.62 paveikslą nustatyta, kad balso srities 1-ojo modelio komponentės 1060 pasikartojimo dažnių įrašuose pirmojo kvartilio reikšmė siekia 1,657%, antrojo – 2,476%, trečiojo – 3,664%.



3.63 pav. Moterų balso srities 1-ojo modelio komponentės 676 pasikartojimo dažnių įrašuose pasiskirstymas

Išanalizavus 3.63 paveiksle pateiktą moterų balso srities 1-ojo modelio komponentės 676 pasikartojimo dažnių įrašuose pasiskirstymą išsiaiškinta, kad komponentės pasikartojimo dažnių įrašuose pirmojo kvartilio reikšmė tik 0,857%, antrojo kvartilio – 1,356%, trečiojo kvartilio – 2,521%.

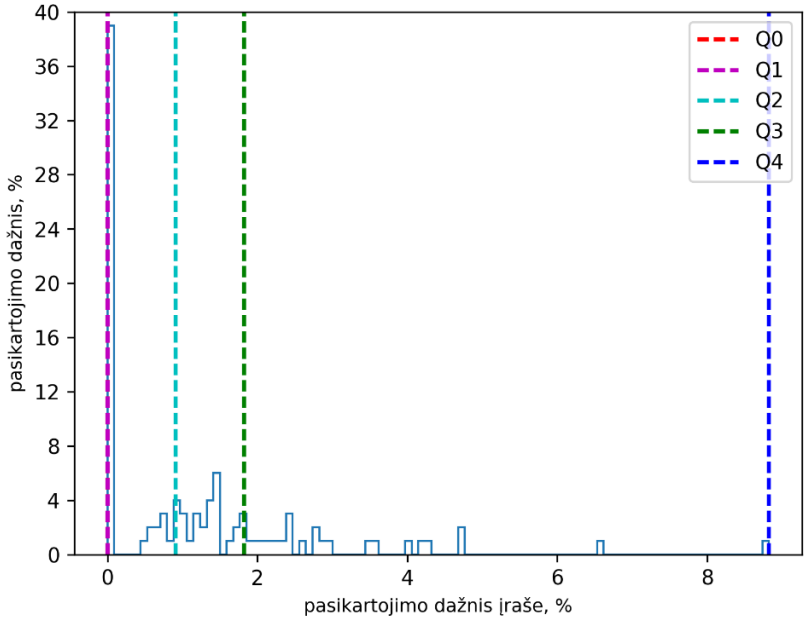
Dažniausiai pasitaikančios vokiečių kalbos garso įrašų moterų balso sričių komponentės nustatytos taikant įvairius akustinius modelius (žr. 3.17 lentelė).

3.17 lentelė. Vokiečių kalbos moterų įrašuose dažniausiai pasitaikančios balso sričių komponentės

Modelis	Balso sričių komponentės	Užima moterų įrašų balso sričių, %
1	548, 1060, 36, 1572, 1348, 1311	10,5
2	1017, 1267, 1733, 674, 725, 1567	5,3
11	0	100
12	1828, 1387, 136, 783, 1724, 1556	6,2

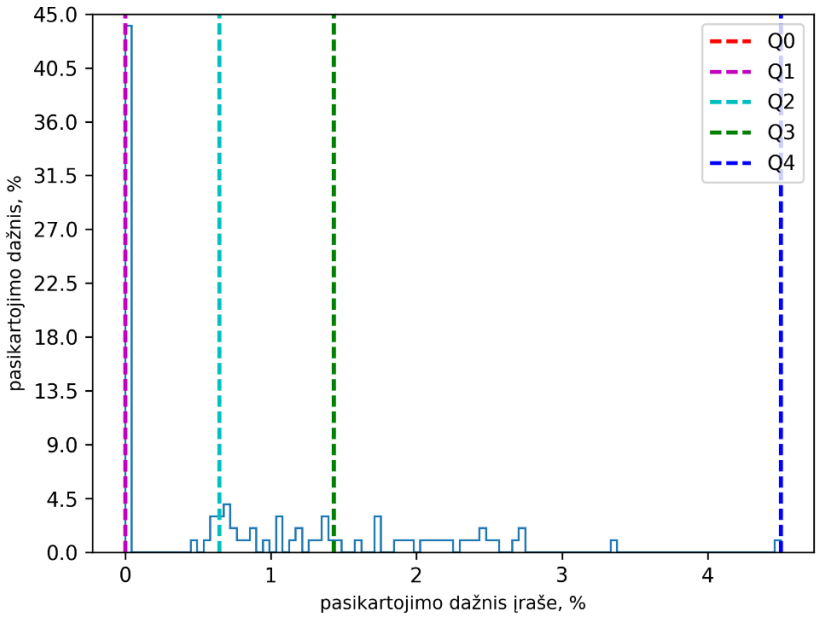
Pagal lentelės rezultatus išsiaiškinta, kad dažniausiai pasitaikančios 11-tojo modelio komponentės nenustatomos. Taip nutiko dėl to, kad tirtuose įrašuose mažai pašalinių triukšmų, todėl vokiečių kalbos garso įrašų lyginamoji analizė atliekama be 11-ojo akustinio modelio rezultatų. Be to nustatyta, kad dažniausiai pasitaikančios moterų balso sričių 2-ojo modelio komponentės užima mažiausią moterų įrašų balso sričių dalį (5,3%). Didžiausią moterų garso įrašų balso sričių dalį užima 1-ojo modelio komponentės (10,5%).

Moterų fono sričių 1-ojo modelio komponenčių 1662, 798 ir 1918 pasikartojimo dažnių įrašuose pasiskirstymai pateikiami 3.64, 3.65 ir 3.66 paveiksluose.



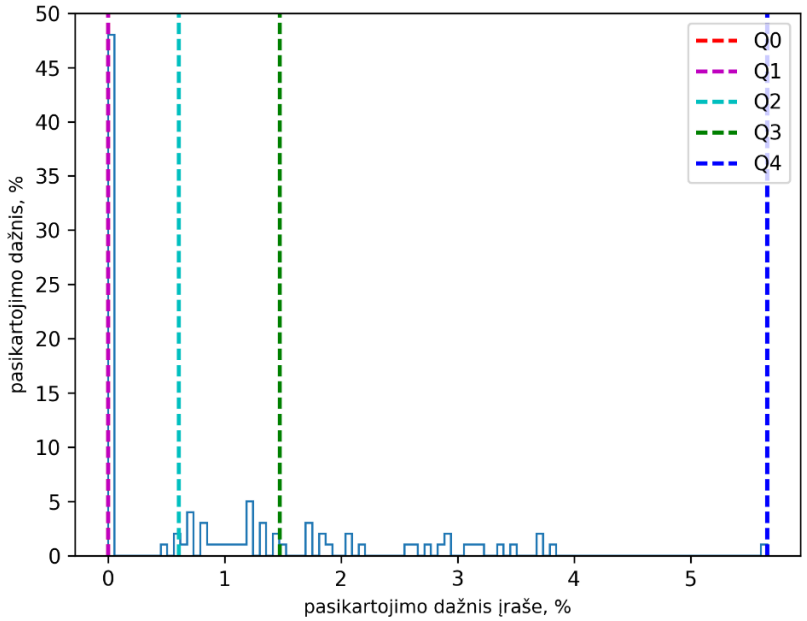
3.64 pav. Moterų fono srities 1-ojo modelio komponentės 1662 pasikartojimo dažnių įrašuose pasiskirstymas

Išnagrinėjus moterų fono srities 1-ojo modelio komponentės 1662 pasikartojimo dažnių įrašuose pasiskirstymą (žr. 3.64 pav.) pastebėta, kad komponentės pirmojo kvartilio reikšmė – tik 0%, antrojo kvartilio – 0,907%, trečiojo – 1,824%.



3.65 pav. Moterų fono srities 1-ojo modelio komponentės 798 pasikartojimo dažnių įrašuose pasiskirstymas

Ištyrus moterų fono srities 1-ojo modelio komponentės 798 pasikartojimo dažnių įrašuose pasiskirstymą (žr. 3.65 pav.) nustatyta, kad komponentės pasikartojimo dažnių įrašuose pirmojo kvartilio reikšmė – 0%, antrojo – 0,647%, trečiojo – 1,433%.



3.66 pav. Moterų fono srities 1-ojo modelio komponentės 1918 pasikartojimo dažnių įrašuose pasiskirstymas

Moterų fono srities 1-ojo modelio komponentės 1918 pasikartojimo dažnių vokiečių kalbos įrašuose pirmojo kvartilio reikšmė taip pat 0%, antrojo – 0,606%, trečiojo – 1,476%.

Taikant įvairius akustinius modelius nustatytos moterų fono srityse dažniausiai pasitaikančios komponentės (žr. 3.18 lentelė).

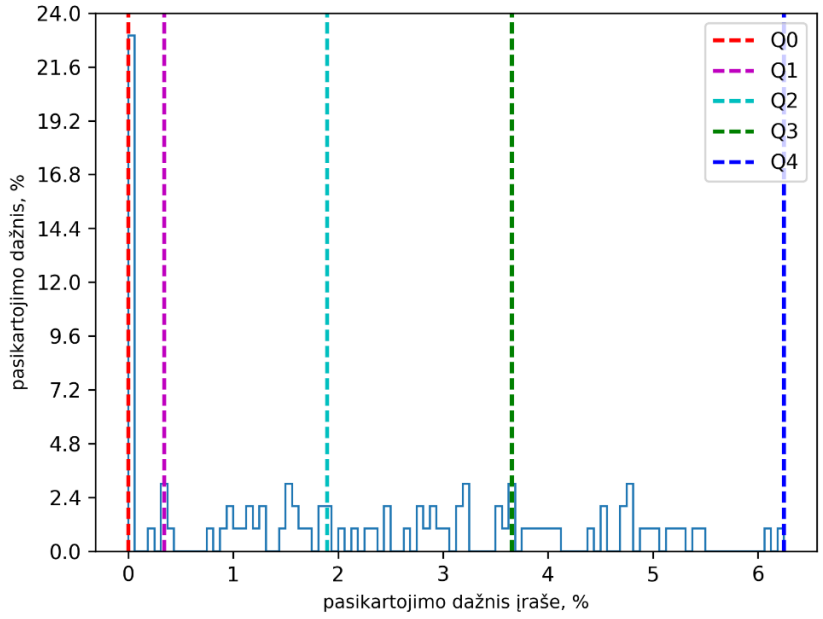
3.18 lentelė. Vokiečių kalbos moterų įrašuose dažniausiai pasitaikančios fono sričių komponentės

Modelis	Fono sričių komponentės	Užima moterų įrašų fono sričių, %
1	1934, 2046, 617, 1529, 899, 1022	13,7
2	1742, 1675, 674, 1303, 1169, 1960	13,7
11	0	100
12	1294, 291, 96, 1052, 570, 788	19,6

Išsiaiškinta, kad dažniausiai pasitaikančios moterų fono sričių 12-ojo modelio komponentės užima daugiausiai moterų garso įrašų fono sričių – 19,6%, o mažiausia dalis užimama 1-ojo ir 2-ojo modelio komponentėmis – po 13,7% moterų įrašų fono sričių.

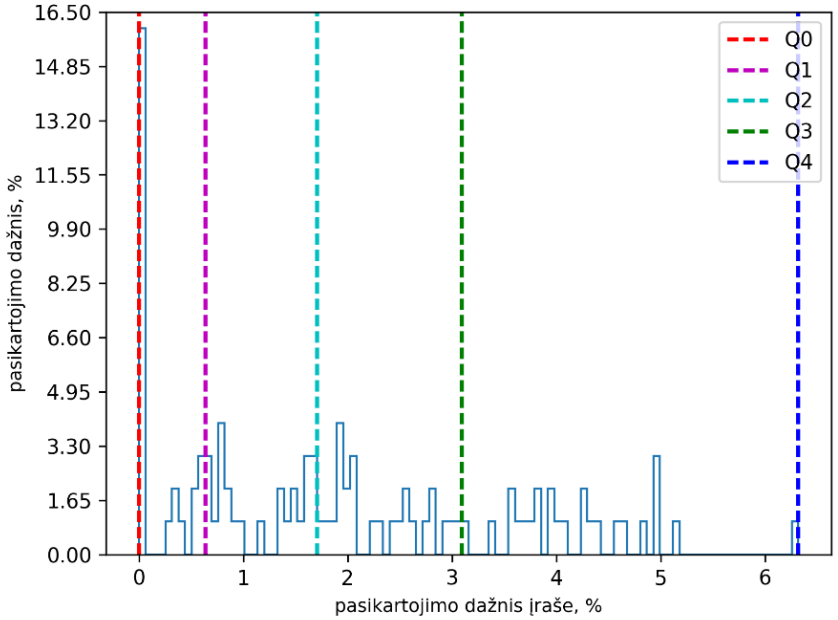
Be to nustatyta, kad dažniausiai pasitaikančios moterų balso ir fono sričių 1-ojo modelio komponentės skiriasi. Toliau pateikiami ir analizuojami vyrų garso įrašų tyrimo metu gauti komponenčių pasikartojimo dažniai, nustatomos dažniausiai pasikartojančios balso ir fono sričių komponentės.

Pagal vyrų garso įrašų tyrimo metu gautus rezultatus sudaryti vyrų balso sričių 1-ojo modelio komponenčių 1060, 548 ir 1311 pasikartojimo dažnių įrašuose pasiskirstymų grafikai.



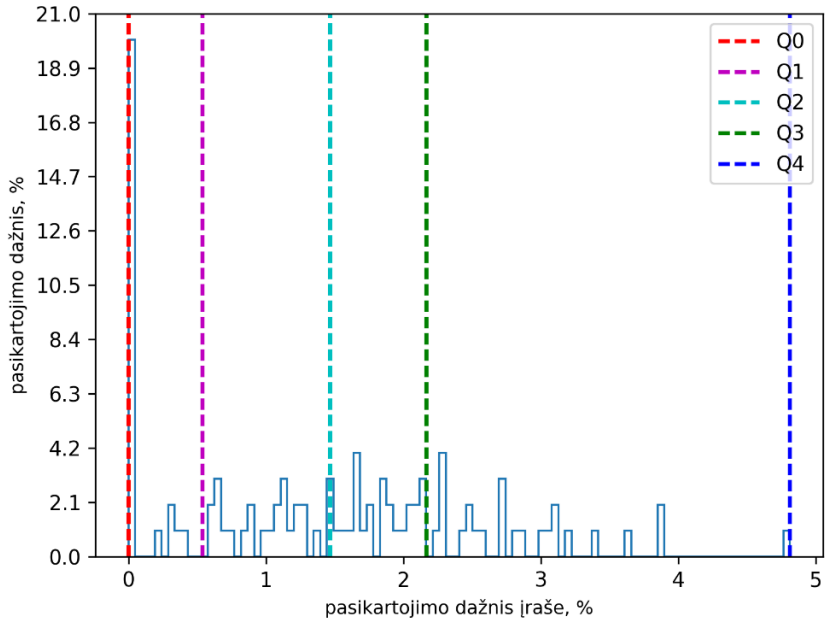
3.67 pav. Vyrų balso srities 1-ojo modelio komponentės 1060 pasikartojimo dažnių įrašuose pasiskirstymas

Ištyrus 3.67 paveiksle pateiktą vyrų balso srities 1-ojo modelio komponentės 1060 pasikartojimo dažnių įrašuose pasiskirstymą atskleista, kad komponentės pirmojo kvartilio reikšmė tik 0,345%, antrojo kvartilio – 1,896%, trečiojo kvartilio – 3,654%.



3.68 pav. Vyrų balso srities 1-ojo modelio komponentės 548 pasikartojimo dažnių įrašuose pasiskirstymas

Išanalizavus 3.68 paveiksle pateiktą vyrų balso srities 1-ojo modelio komponentės 548 pasikartojimo dažnių įrašuose pasiskirstymą pastebėta, kad komponentės pirmojo kvartilio reikšmė – 0,637%, antrojo – 1,709%, trečiojo – 3,094%.



3.69 pav. Vyrų balso srities 1-ojo modelio komponentės 1311 pasikartojimo dažnių įrašuose pasiskirstymas

Išnagrinėjus vyrų balso srities 1-ojo modelio komponentės 1311 pasikartojimo dažnių įrašuose pasiskirstymą (žr. 3.69 pav.) išsiaiškinta, kad komponentės pirmojo kvartilio reikšmė – 0,54%, antrojo kvartilio – 1,468%, trečiojo – 2,169%.

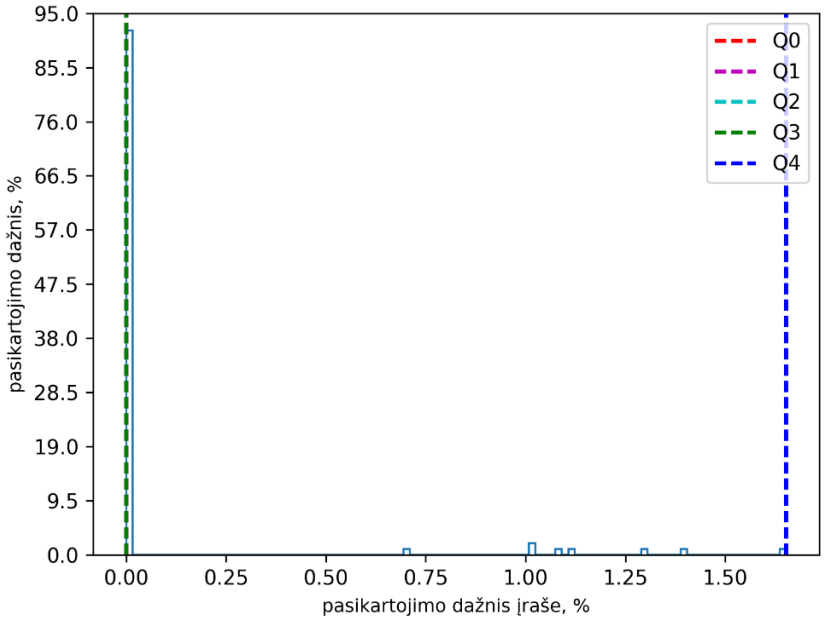
Dažniausiai pasitaikančios vyrų balso sričių komponentės nustatytos taikant įvairius akustinius modelius (3.19 paveiksle).

3.19 lentelė. Vokiečių kalbos vyrų įrašuose dažniausiai pasitaikančios balso sričių komponentės

Modelis	Balso sričių komponentės	Užima vyrų įrašų balso sričių, %
1	377, 1311, 1060, 2041, 1695, 1607	14,2
2	41, 1365, 987, 1291, 677, 508	7,1
11	0	100
12	1389, 1365, 1341, 1282, 375, 1676	11,6

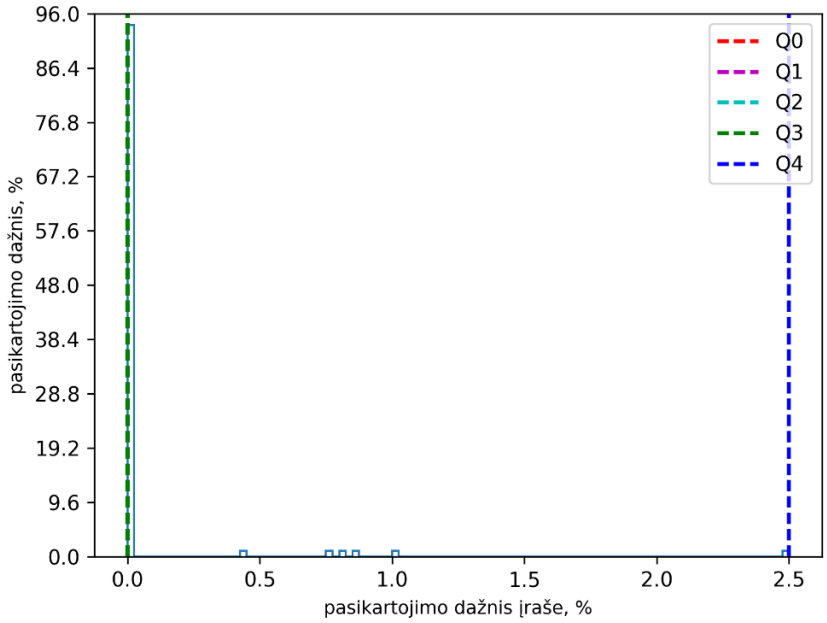
Išsiaiškinta, kad dažniausiai pasitaikančios vyrų balso sričių 2-ojo modelio komponentės užima mažiausią vyrų garso įrašų balso sričių dalį (7,1%). Didžiausia vyrų įrašų balso sričių dalį užima 1-ojo modelio komponentės (14,2%).

Vyrų fono sričių 1-ojo modelio komponenčių (2042, 667, 722) pasikartojimo dažnių vokiečių kalbos įrašuose pasiskirstymai vaizduojami 3.70, 3.71 ir 3.72 paveiksluose.



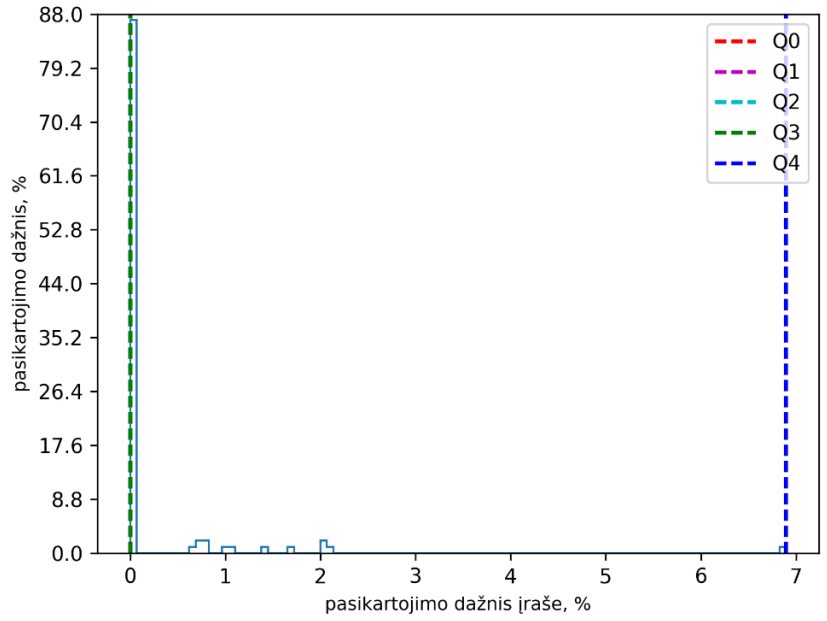
3.70 pav. Vyrų fono srities 1-ojo modelio komponentės 2042 pasikartojimo dažnių įrašuose pasiskirstymas

Ištyrus vyrų fono srities 1-ojo modelio komponentės 2042 pasikartojimo dažnių įrašuose pasiskirstymą (žr. 3.70 pav.) išsiaiškinta, kad komponentės pirmojo, antrojo ir trečiojo kvartilių reikšmės tokios pat – 0%.



3.71 pav. Vyrų fono srities 1-ojo modelio komponentės 667 pasikartojimo dažnių įrašuose pasiskirstymas

Išanalizavus 3.71 paveikslą išsiaiškinta, kad vyrų fono srities 1-ojo modelio komponentės 667 pasikartojimo dažnių įrašuose pirmojo, antrojo ir trečiojo kvartilių reikšmės vienodos (0%).



3.72 pav. Vyrų fono srities 1-ojo modelio komponentės 722 pasikartojimo dažnių įrašuose pasiskirstymas

Išnagrinėjus vyrų fono srities 1-ojo modelio komponentės 722 pasikartojimo dažnių įrašuose pasiskirstymą išsiaiškinta, kad komponentės pirmojo, antrojo ir trečiojo kvartilių reikšmės vėlgi – 0%.

Įvairiais akustiniais modeliais nustatytos vyrų fono srityse dominuojančios komponentės ir jų užimamos vyrų įrašų fono sričių dalys (žr. 3.20 lentelė).

3.20 lentelė. Vokiečių kalbos vyrų įrašuose dažniausiai pasitaikančios fono sričių komponentės

Modelis	Fono sričių komponentės	Užima vyrų įrašų fono sričių, %
1	440, 1825, 1282, 829, 1037, 1193	14,6
2	893, 190, 797, 3, 1276, 1261	30,8
11	0	100
12	1435, 1937, 1857, 1792, 1580, 678	45,4

Pagal lentelės duomenis išsiaiškinta, kad dažniausiai pasitaikančios vyrų fono sričių komponentės užima daugiausiai vyrų įrašų fono sričių taikant 12-tąjį modelį – 37,6%, mažiausiai užima 1-ojo modelio komponentės – 14,6%.

Apibendrinus galima teigti, kad komponentės įrašuose pasiskirsto nevienodai. Balso ir fono sričių komponenčių rezultatų analizė atskleidė, kad pasirinktas tinkamas normavimo būdas, kadangi balso sričių komponentės nėra susimaišiusios su fono sričių komponentėmis, todėl balso sričių komponentės atitinka kalbos garsus.

3.1.7. Kalbų palyginimas

Įvairių akustinių modelių komponenčių statistikų tyrimo metu nustatyta, kad komponentės pasiskirsto įrašuose nevienodai. Įvertinant modelius išrinktos 6 dažniausiai pasitaikančios akustinių modelių komponentės. Įvairių kalbų įrašuose dažniausiai pasitaikančios balso sričių anglų kalbos akustinių modelių komponentės pateiktos 3.21 lentelėje.

3.21 lentelė. Dažniausiai pasitaikančios balso sričių anglų kalbos akustinių modelių komponentės

Įrašai	1-ojo modelio komponentės	2-ojo modelio komponentės
Anglų moterų	548, 1060, 676, 1188, 1700, 164	556, 210, 804, 326, 1250, 379
Anglų vyrų	548, 1060, 676, 1311, 164, 1700	1959, 1809, 1822, 397, 118, 1993
Ispanų moterų	1420, 151, 1444, 772, 87, 1022	1335, 822, 1226, 1040, 1912, 910
Ispanų vyrų	1188, 932, 244, 1060, 1700, 548	3, 77, 773, 793, 807, 1615
Italų moterų	4, 84, 603, 1845, 1980, 1027	829, 1012, 311, 865, 1360, 1275
Italų vyrų	676, 164, 1420, 548, 1311, 1060	707, 915, 1850, 1344, 65, 812
Prancūzų moterų	1060, 548, 1572, 1311, 1796, 676	1529, 246, 1670, 1335, 1234, 1031
Prancūzų vyrų	548, 1060, 1311, 36, 676, 151	661, 832, 363, 1988, 1978, 763
Rusų moterų	548, 1060, 164, 1188, 1444, 953	1554, 815, 1782, 1939, 143, 1993
Rusų vyrų	1311, 377, 889, 1444, 420, 932	1039, 1299, 1037, 581, 599, 773
Vokiečių moterų	548, 1060, 36, 1572, 1348, 1311	1017, 1267, 1733, 674, 725, 1567
Vokiečių vyrų	377, 1311, 1060, 2041, 1695, 1607	41, 1365, 987, 1291, 677, 508

Pagal lentelės rezultatus išsiaiškinta, kad 6 dažniausiai pasikartojančios anglų kalbos GMM modelio komponentės užima 12,5% balso sričių požymių moterų įrašuose anglų kalba, 9% – vyrų garso įrašuose. 5 iš šių komponenčių yra bendros. Dažniausiai pasikartojanti 1-ojo modelio komponentė

(1060) taipogi bendra. Tokią pat poziciją ji užima vyrų įrašuose – prancūzų ir anglų kalbomis, moterų įrašuose – anglų, rusų bei vokiečių kalbomis. Daugumoje kitų tirtų kalbų ji patenka į dažniausių komponenčių šešetą. Be to išsiaiškinta, kad k-vidurkių metode tokio dėsningumo nėra. Toliau aptariami įvairių kalbų akustinių modelių, skirtų kalbėtojų atpažinimui, patikrinimo rezultatai.

3.2. Įvairių kalbų akustinių modelių, skirtų kalbėtojų atpažinimui, patikrinimo rezultatai

Siekiant sudaryti įvairių kalbų akustinius modelius, atliktas tyrimas, kurio metu išskirti anglų, ispanų, italų, prancūzų, rusų ir vokiečių kalbų garso įrašų kalbos požymiai taikyti tų kalbų akustinių modelių sudarymui. Kalbos signalų akustinių modelių sudarymui taikytos akustinių modelių sudarymo GMM (žr. 2.2 pav.) ir k-vidurkių klasterizavimo metodu (žr. 2.3 pav.) pseudokodų logikos. Sudaryti modeliai tikrinti atsitiktinai parinktais testavimo įrašais. Patikrinimui taikyta po 10 kiekvienos kalbos garso įrašų, iš kurių atsitiktinai parinkta po 5 moterų ir po 5 vyrų garso įrašus.

Kad būtų nustatyta, kurie akustiniai modeliai tinkamiausi – taikant įvairius akustinius modelius kiekvienai nagrinėtai kalbai apskaičiuoti įrašų logaritminiai tikėtinumai. Gautiems rezultatams atlikta statistinė analizė: vieno veiksnio dispersinė analizė ir Stjudento t-testas.

Siekiant išsiaiškinti, ar skirtingų akustinių modelių logaritminių tikėtinumų rezultatai statistiškai reikšmingai skiriasi nuo kalbos, atlikta vieno veiksnio dispersinė analizė, kur veiksniu parinkta kalba, o reikšmingumo lygmuo – 0,05.

Įvairių kalbų garso įrašų logaritminiai tikėtinumai, gauti taikant 1-ąjį akustinį modelį, pateikiami 3.22 lentelėje.

3.22 lentelė. Įvairių kalbų įrašų logaritminiai tikėtinumai taikant 1-ąjį modelį

Įrašo eil. nr.	1 modelio logaritminiai tikėtinumai					
	Anglų k.	Ispanų k.	Italų k.	Prancūzų k.	Rusų k.	Vokiečių k.
1	-74,25	-75,85	-74,16	-69,28	-72,14	-71,09
2	-67,99	-95,29	-86,33	-71,90	-70,75	-79,68
3	-78,12	-94,65	-76,03	-68,60	-64,20	-72,28
4	-72,12	-108,23	-88,59	-70,54	-69,47	-66,40
5	-66,47	-73,51	-64,07	-73,66	-80,92	-74,73
6	-68,04	-80,43	-67,84	-72,71	-74,28	-68,44
7	-76,14	-70,67	-85,49	-67,40	-67,83	-76,78
8	-68,57	-67,44	-67,90	-65,54	-72,25	-79,12
9	-74,65	-70,30	-72,69	-72,86	-71,09	-77,17
10	-76,42	-66,41	-76,87	-65,63	-70,36	-85,03

Nagrinėta, ar įvairių kalbų garso įrašų logaritminiai tikėtinumai statistiškai reikšmingai skiriasi, taikant 1-ąjį akustinį modelį. Iškeltos statistinės hipotezės:

- H_0: įvairių kalbų garso įrašų logaritminiai tikėtinumai statistiškai reikšmingai nesiskiria, taikant 1-ąjį akustinį modelį.
- H_1: įvairių kalbų garso įrašų logaritminiai tikėtinumai statistiškai reikšmingai skiriasi, taikant 1-ąjį modelį.

Gauta, kad p=0,045<α=0,05, todėl, įvairių kalbų garso įrašų logaritminiai tikėtinumai statistiškai reikšmingai skiriasi, taikant 1-ąjį modelį, esant 0,05 reikšmingumo lygmeniui.

Nagrinėjant įvairių kalbų įrašų 1-ojo modelio logaritminius tikėtinumus pastebėta, kad ispanų kalbos rezultatai labai skiriasi nuo kitų kalbų. Dėl to analizė pakartota be šios kalbos. Gauta, kad p=0,08>α=0,05. Vadinasi, iš analizės pašalinus ispanų kalbą, gaunama, kad įvairių kalbų garso įrašų logaritminiai tikėtinumai statistiškai reikšmingai nesiskiria, taikant 1-ąjį akustinį modelį esant 0,05 reikšmingumo lygmeniui.

Įvairių kalbų įrašų logaritminiai tikėtinumai gauti taikant 2-ąjį modelį suvesti 3.23 lentelėje.

3.23 lentelė. Įvairių kalbų įrašų logaritminiai tikėtinumai taikant 2-ąjį modelį

Įrašo eil. nr.	2 modelio logaritminiai tikėtinumai					
	Anglų k.	Ispanų k.	Italų k.	Prancūzų k.	Rusų k.	Vokiečių k.
1	-78,57	-79,77	-78,05	-75,91	-77,17	-76,56
2	-74,94	-94,35	-86,22	-78,42	-76,44	-81,62
3	-81,14	-93,59	-79,63	-76,50	-72,67	-76,82
4	-78,19	-103,77	-89,99	-76,28	-76,20	-73,80
5	-73,76	-77,95	-73,25	-78,17	-82,39	-78,54
6	-74,84	-82,55	-74,17	-77,70	-78,64	-75,09
7	-79,43	-76,55	-85,59	-75,26	-74,83	-80,50
8	-74,91	-74,69	-75,01	-73,77	-77,61	-81,26
9	-78,50	-76,47	-77,63	-78,32	-76,88	-80,52
10	-80,48	-73,99	-79,95	-80,48	-76,16	-84,89

Siekiant apibendrinti 3.22 lentelės rezultatus iškeltos statistinės hipotezės.

- H_0: įvairių kalbų garso įrašų logaritminiai tikėtinumai statistiškai reikšmingai nesiskiria, taikant 2-ąjį akustinį modelį.
- H_1: įvairių kalbų garso įrašų logaritminiai tikėtinumai statistiškai reikšmingai skiriasi, taikant 2-ąjį modelį.

Gauta, kad p=0,083>α=0,05, todėl įvairių kalbų garso įrašų logaritminiai tikėtinumai statistiškai reikšmingai nesiskiria, taikant 2-ąjį akustinį modelį esant 0,05 reikšmingumo lygmeniui.

Ispanų, italų, prancūzų, rusų ir vokiečių kalbų GMM modelių sudarymui taikyti tų kalbų įrašų tyrimo metu išskirti balso sričių požymiai. Tokiu būdų sudaryti 3-iasis, 5-asis, 7-asis, 9-asis ir 11-asis modeliai. Nustatyti įvairių kalbų logaritminiai tikėtinumai, taikant tos kalbos GMM modelį (žr. 3.24 lentelė).

3.24 lentelė. Įvairių kalbų įrašų logaritminiai tikėtinumai taikant tos kalbos GMM modelį

Įrašo eil. nr.	Logaritminiai tikėtinumai				
	Ispanų k.	Italų k.	Prancūzų k.	Rusų k.	Vokiečių k.
	3 modelis	5 modelis	7 modelis	9 modelis	11 modelis
1	-75,51	-76,87	-72,07	-70,03	-72,89
2	-96,88	-91,67	-76,94	-68,73	-84,56
3	-94,29	-78,56	-73,34	-59,56	-73,80
4	-107,02	-97,95	-74,23	-67,47	-69,26
5	-71,89	-65,86	-74,76	-80,02	-78,43
6	-78,52	-69,49	-75,29	-72,38	-70,22
7	-66,29	-91,53	-70,68	-62,94	-84,32
8	-61,95	-70,77	-70,10	-69,85	-81,92
9	-66,00	-78,31	-79,73	-68,86	-83,95
10	-60,52	-81,19	-68,32	-66,48	-93,78

Pagal įvairių kalbų GMM modelių logaritminių tikėtinumų reikšmes atlikta statistinė analizė:

- H_0: įvairių kalbų įrašų logaritminiai tikėtinumai statistiškai reikšmingai nesiskiria, taikant tų kalbų GMM modelius.
- H_1: įvairių kalbų įrašų logaritminiai tikėtinumai statistiškai reikšmingai skiriasi, taikant tų kalbų GMM modelius.

Gauta, kad p=0,061>α=0,05. Taigi, įvairių kalbų įrašų logaritminiai tikėtinumai statistiškai reikšmingai nesiskiria, taikant tų kalbų GMM modelius esant 0,05 reikšmingumo lygmeniui.

Kad būtų sudaryti ispanų, italų, prancūzų, rusų ir vokiečių kalbų k-vidurkių modeliai, taikyti tų kalbų garso įrašų balso sričių požymiai. Sudaryti 4-asis, 6-asis, 8-asis, 10-asis ir 12-asis kalbos signalų akustiniai modeliai. Taikant sudarytus tų kalbų k-vidurkių akustinius modelius nustatyti įvairių kalbų logaritminiai tikėtinumai (žr. 3.25 lentelė).

3.25 lentelė. Įvairių kalbų įrašų logaritminiai tikėtinumai taikant tos kalbos k-vidurkių modelį

Įrašo eil. nr.	Logaritminiai tikėtinumai				
	Ispanų k.	Italų k.	Prancūzų k.	Rusų k.	Vokiečių k.
	4 modelis	6 modelis	8 modelis	10 modelis	12 modelis
1	-79,52	-78,77	-81,88	-79,62	-81,58
2	-84,79	-83,50	-81,16	-78,00	-84,76
3	-85,39	-79,65	-80,11	-78,69	-82,95
4	-96,60	-82,70	-82,96	-79,13	-81,51
5	-80,54	-77,75	-82,73	-83,54	-84,36
6	-80,62	-79,44	-83,42	-80,33	-81,46
7	-78,46	-82,99	-81,22	-78,63	-83,03
8	-78,62	-77,87	-80,14	-80,34	-84,43
9	-77,64	-78,41	-81,98	-77,62	-83,06
10	-76,84	-79,00	-80,66	-79,48	-85,58

Apskaičiavus įvairių kalbų k-vidurkių modelių logaritminių tikėtinumų reikšmes iškeltos statistinės hipotezės:

- H_0: įvairių kalbų įrašų logaritminiai tikėtinumai statistiškai reikšmingai nesiskiria, taikant tų kalbų k-vidurkių modelius.
- H_1: įvairių kalbų įrašų logaritminiai tikėtinumai statistiškai reikšmingai skiriasi, taikant tų kalbų k-vidurkių modelius.

Gauta, kad p=0,057>α=0,05. Vadinasi, įvairių kalbų įrašų logaritminiai tikėtinumai statistiškai reikšmingai nesiskiria, taikant tų kalbų k-vidurkių modelius esant 0,05 reikšmingumo lygmeniui.

Atlikus vieno veiksnio dispersinę įvairių kalbų garso įrašais išsiskinta, jog skirtinų kalbų įrašų logaritminiai tikėtinumai statistiškai reikšmingai nesiskiria, taikant tų kalbų to paties tipo modelį.

Siekiant nustatyti, ar įrašų logaritminiai tikėtinumai lyginamoms kalboms statistiškai reikšmingai skiriasi, taikant anglų kalbos modelius, atliktas Stjudento t-testas, kur lyginamos kalbos – tai anglų ir kita (ispanų, italų, prancūzų, rusų arba vokiečių) testavimo kalba, o reikšmingumo lygmuo – 0,05. Iškeltos hipotezės:

- H_0: įrašų logaritminiai tikėtinumai statistiškai reikšmingai nesiskiria lyginamoms kalboms, taikant anglų kalbos akustinius modelius.
- H_1: įrašų logaritminiai tikėtinumai statistiškai reikšmingai skiriasi lyginamoms kalboms, taikant anglų kalbos akustinius modelius.

Stjudento t-testo anglų kalbos akustinių modelių rezultatai suvesti 3.26 lentelėje.

3.26 lentelė. Anglų kalbos akustinių modelių Stjudento t-testo rezultatai

Lyginamos kalbos	**p reikšmė**	
	1 modelis	**2 modelis**
anglų/ ispanų	0,118	0,108
anglų/ italų	0,235	0,230
anglų/ prancūzų	0,151	1,000
anglų/ rusų	0,627	0,623
anglų/ vokiečių	0,227	0,291

Gauta, kad įrašų logaritminiai tikėtinumai statistiškai reikšmingai nesiskiria lyginamoms kalboms, taikant 1-ąjį ir 2-ąjį modelius, kadangi gautos p reikšmės viršija pasirinktą reikšmingumo lygmenį.

Siekiant išsiaiškinti, kurios kalbos logaritminių tikėtinumų reikšmės labiausiai skiriasi nuo anglų kalbos, atliktas tyrimas, kurio metu, taikant anglų kalbos akustinius modelius, apskaičiuoti įvairių kalbų garso įrašų 1-ojo ir 2-ojo akustinių modelių logaritminių tikėtinumų vidurkiai.

Pagal 3.21 lentelės duomenis apskaičiuotos įvairių kalbų įrašų 1-ojo akustinio modelio logaritminių tikėtinumų vidurkių reikšmės: ispanų – -80,28, italų – -76, vokiečių – -75,07, anglų – -72,28, rusų – -71,33, prancūzų – -69,81.

Taikant 3.22 lentelės rezultatus nustatytos įvairių kalbų garso įrašų 2-ojo akustinio modelio logaritminių tikėtinumų vidurkių reikšmės: ispanų kalbos – -83,37, italų – -79,95, vokiečių – -78,96, anglų – -77,48, rusų – -76,9, prancūzų – -76,37.

Kadangi ispanų kalbos įrašų 1-ojo ir 2-ojo modelio logaritminių tikėtinumų vidurkių reikšmės mažiausios, vadinasi ispanų ir anglų kalbų įrašų logaritminiai tikėtinumai labiausiai skiriasi. Dėl to atsitiktinai parinkta dar po 100 anglų ir ispanų kalbos testavimo įrašų, taikant anglų kalbos akustinius modelius apskaičiuoti jų logaritminiai tikėtinumai. Pagal gautus rezultatus atliktas Stjudento t-testas, kur lyginamoms kalboms (anglų ir ispanų) iškeltos statistinės hipotezės:

- H_0: padidinus įrašų imtį logaritminiai tikėtinumai statistiškai reikšmingai nesiskiria anglų ir ispanų kalboms, taikant anglų kalbos akustinius modelius.
- H_1: padidinus įrašų imtį logaritminiai tikėtinumai statistiškai reikšmingai skiriasi anglų ir ispanų kalboms, taikant anglų kalbos akustinius modelius.

Gauta, kad 1-ojo modelio $p=0,000<\alpha=0,05$ ir 2-ojo modelio $p=0,000<\alpha=0,05$. Vadinasi, padidinus įrašų imtį, logaritminiai tikėtinumai statistiškai reikšmingai skiriasi anglų ir ispanų kalboms, taikant anglų kalbos akustinius modelius esant 0,05 reikšmingumo lygmeniui.

IŠVADOS

Mokslinės literatūros analizės metu buvo išsiaiškinta, kad:

- kalbėtojo atpažinimas – nuolat besivystanti biometrijos rūšis, kuri tobulinama kuriant naujus algoritmus pradedant nuo paslėptųjų Markovo modelių ir baigiant ties moderniaisiais konvoliuciniais tinklais ar LSTM.

Kalbos signalų akustinių modelių, skirtų kalbėtojo atpažinimui, tyrimo metu išsiaiškinta, kad:

- komponentės pasiskirsto įrašuose nevienodai. Įvertinant modelius išrinktos 6 dažniausiai pasitaikančios modelių komponentės. Anglų kalbos GMM modelio 6 vyraujančios komponentės užima 12,5% balso sričių požymių moterų įrašuose, 9% – vyrų įrašuose anglų kalba. 5 iš šių komponenčių yra bendros. Dažniausiai pasikartojanti komponentė taip pat yra bendra. Tokį pat vaidmenį ji turi įrašuose: vyrų – prancūzų kalba, moterų – rusų bei vokiečių kalbomis. Daugumoje kitų tirtų kalbų ji patenka į dažniausių komponenčių šešetą;
- dažniausiai pasitaikančios balso ir fono srityse komponentės yra skirtingos;
- atlikus vieno veiksnio dispersinę analizę gauta, kad skirtinų kalbų įrašų logaritminiai tikėtinumai statistiškai reikšmingai nesiskiria, taikant tų kalbų to paties tipo modelį;
- atlikus anglų kalbos akustinių modelių logaritminių tikėtinumų vidurkių analizę nustatyta, kad ispanų ir anglų kalbų įrašų logaritminiai tikėtinumai labiausiai skiriasi;
- Stjudento t-testo metu gauta, kad įrašų logaritminiai tikėtinumai nuo įrašų kalbų statistiškai reikšmingai nesiskiria, taikant anglų kalbos akustinius modelius. Padidinus įrašų imtį, logaritminiai tikėtinumai statistiškai reikšmingai skiriasi anglų ir ispanų kalboms, taikant anglų kalbos akustinius modelius.

LITERATŪRA

1. *A practical approach to Automatic Speech Recognition using Deep Learning* [interaktyvus] [žiūrėta 2017-04-17]. Prieiga per internetą: < http://algomuse.com/python/a-practical-approach-to-automatic-speech-recognition-using-deep-learning>.
2. ATAL, B. S., HANAUER, S. L. *Speech Analysis and Synthesis by Linear Prediction of the Speech Wawe*. J. Acoust Soc. Am., 1971, vol. 50, no. 2, 637-655 p.
3. *Balso technologijų panaudojimo kriminalistikoje analizė pasaulyje ir Lietuvoje* [interaktyvus] [žiūrėta 2016-06-08]. Prieiga per internetą: <http://www.likit.lt/all/balso_tech/05_kriminal.htm>.
4. CHAUDHARI, P. R., ALEX, J. S. R. *Low power, small foot print embedded voice biometrics system*. Pak. J. Biotechnol. 2016, vol. 13, 121-124 p.
5. COLLOBERT, R., PUHRSCH, C., SYNNAEVE, G. *Wav2Letter: an End-to-End ConvNet-based Speech REcognition System*. Facebook All Research., 2016, 1-8 p.
6. *Convolutional Neural Networks* [interaktyvus] [žiūrėta 2016-03-28]. Prieiga per internetą: <http://andrew.gibiansky.com/blog/machine-learning/convolutional-neural-networks/>.
7. DEHAK, N., RICHARDSON, F., REYNOLDS, D. *A unified deep neural network for speaker and language recognition*. Interspeech, Germany, 2015, 1146-1150 p.
8. DURRETT R. *Probability: Theory and Examples*. New York: Cambridge University Press. 2013, p. 386.
9. GAIDA, C., LANGE, P., PETRICK, R. ir kt. *Comparing Open-Source Speech Recognition Toolikts*. Technical Report of the Project OASIS, DHBW Stuttgart, Germany, 2014.
10. GALVEZ, D. *GITHUB: acc-lda.cc* [interaktyvus] [žiūrėta 2016-09-30]. Prieiga per internetą: <https://github.com/kaldi-asr/kaldi/blob/master/src/bin/acc-lda.cc>.
11. GRAVES, A. *Supervised Sequence Labelling with Recurrent Neural Network*. Springer, 2012, 137 p.
12. *Hidden Markov Models* [interaktyvus] [žiūrėta 2016-03-02]. Prieiga per internetą: <http://digital.cs.usu.edu/~cyan/CS7960/hmm-tutorial.pdf>.
13. ITAKURA, F. *Minimum prediction Residual Applied to Speech Recognition*. IEEE Trans. Acoustics, Speech signal Proc., 1975, vol. ASSP-23, no. 1, 67-72 p.
14. YOUNG, S. J., WOODLAND, P. C., BYRNE, W. J. *The HTK Book (for HTK Version 3.5, documentation alpha versijon)*. UK: Cambridge University Engineering Department, 2015, p. 355.

15. YU, D., DENG, L. *Automatic Speech Recognition: A Deep Learning Approach*. Springer London, 2015, p. 315.
16. *Kaldi* [interaktyvus] [žiūrėta 2016-07-23]. Prieiga per internetą: <http://kaldi-asr.org/doc/>.
17. KANAGANSUNDARAM, A., VOGT, R. ir kt. *i-vector Based Speaker Recognition on Short Utterances*. Interspeech, Italy, 2011, 2341-2344 p.
18. LIPEIKA, A. *Signalų ir sistemų dažninė analizė*. Technika, 2003, p. 156.
19. MAESA, A., GARZIA, F., SCARPINITI, M., CUSANI, R. *Text Independent Automatic Speaker Recognition System Using Mel-Frequency Cespstrum Coefficient and Gaussian Mixture Models*. Journal Of Information Security, 2012, vol. 3, 335-340 p.
20. MAHBOOB, T., KHANUM, M., KHIYAL, M. S. H., BIBI, R. *Speaker Identification Using GMM with MFCC*. IJCSI, 2015, vol. 12, no. 2, 126-135 p.
21. MARHAV, N., LEE, C.-H. *On the asymptotic statistical behavior of empirical cepstral coefficients*. IEEE Transactions on Signal Processing, 1993, vol. 41, 1990-1993 p.
22. MONTE-MORENO, E., CHETOUANI, M., FAUNDEZ-ZANUY, M., SOLE-CASALS, J. *Maximum likelihood linear programming data fusion for speaker recognition*. Speech Communication archive, 2008, vol. 51, no. 9, 820-830 p.
23. *Paslėpti Markovo modeliai* [interaktyvus] [žiūrėta 2016-03-02]. Prieiga per internetą: <http://www2.el.vgtu.lt/ssa/sB1node1.html>.
24. PATEL, S. *A lower-complexity Viterbi algorithm*. International Conference on acoustics, Speech, and Signal Processing ICASSP-95, 1995, vol. 1, p. 592-595.
25. POVEY, D. *GITHUB* [interaktyvus] [žiūrėta 2016-09-30]. Prieiga per internetą: < https://github.com/danpovey>.
26. POVEY, D., GHOSHAL, A., BOULIANNE, G., BURGER, L. ir kt. *The Kaldi Speech Recognition Toolkit*. IEEE 2011 Workshop on Automatic Speech Recognition and Understanding, 2011.
27. RABINER, L.R. *A tutorial on hidden Markov models and selecten applications in speech recognition*. Proceeding of the IEEE, 1989, vol. 77, no. 2, 257-286 p.
28. RABINER, L.R., JUANG B.H. *Fundamentals of Speech Recognition*. Prentice-Hall, 1993, p. 507.
29. RAJESWARA RAO, R., NAGESH, A. ir kt. *Text-Dependent Speaker Recognition System for Indian Languages*. International Journal of Computer Science and Network Security, 2007, vol. 7, no. 11, 65-71 p.
30. RAŠKINIS, G., RAŠKINIENĖ, D. *Development of Medium-ocablary Isolated-Word Lithuanian HMM Speech Recognition System*, Vilnius, Informatica, 2003, vol.14, no. 1, 75-84 p.

31. REYNOLDS, D. A. *Automatic Speaker Recognition Using Gaussian Mixture Speaker Models*. The Lincoln Laboratory Journal, 1995, vol. 8, 173-174 p.
32. RUZGYS, T. *Daugiamačio pasiskirstymo tankio neparametrinis įvertinimas naudojant stebėjimų klasterizavimą.* Daktaro disertacija, Vilnius, Matematikos ir informatikos institutas, 2007, p. 94.
33. SADJADI, S. O., SLANEY, M., HECK, L. *MSR Identity Toolbox v1.0: A MATLAB Toolbox for Speaker Recognition Research*. IEEE SLTC Newsletter, 2013.
34. SALMAN, R. Contributions to k-means Clustering and Regression via Classification Algorithms. Dissertation, Virginia Commonwealth University, Department of Computer Science, 2012, p. 98.
35. SNYDER, D., GARCIA-ROMERO, D., POVEY, D. *Time delay deep neural network-based universal background models for speaker recognition*. Proceedings of IEEE Automatic Speech Recognition and Understanding Workshop (ASRU), Scottsdale, Arizona, USA, 2015.
36. SOUZA, C. Gaussian Mixture Models and Expectation-Maximization [interaktyvus] [žiūrėta 2017-04-23]. Prieiga per internetą:<http://crsouza.com/2010/10/18/gaussian-mixture-models-and-expectation-maximization/>.
37. *Speech Recognition with Kaldi Lectures* [interaktyvus] [žiūrėta 2016-09-30]. Prieiga per internetą: <http://danielpovey.com/kaldi-lectures.html>.
38. TIWARI, V. *MFCC and its applications in speaker recognition*. International Journal on Emerging Technologies, 2010, vol. 1, no. 1, 19-22 p.
39. *Types of biometrics* [interaktyvus] [žiūrėta 2016-07-23]. Prieiga per internetą: <http://www.biometricsinstitute.org/pages/types-of-biometrics.html>.
40. VASQUEZ, D., GRUHN, R., MINKER, W. *Hierarchical Neural Network Stuctures for Phoneme Recognition*. Springer: Signals and Communication Technology, 2013, p. 122.
41. VITERBI, A. *Error for convolutional codes and an asymptotically optimum decoding algorithm*. IEEE Transactions on Information Theory, 1967, vol. 13, no. 2, 260-269 p.
42. WANG, D. *Electronic Engineering and Information Science*. CRC Press: Harbin University of Science and Technology, 2015, p. 790.
43. ZHANG, J., STEPHENS, M. A. *A new and efficient estimation method for the generalized Pareto distribution*. Technometrics, 2009, vol. 51, no. 3, 316-325 p.
44. ZIOLKO, B., ZIOLKO, M. ir kt. *Hybrid Wavelet-Fourier-HMM Speaker Recognition*. International Journal of Hybrid Information Technology, 2011, vol. 4, no. 4, 25-42 p.

PRIEDAI

1 PRIEDAS

POŽYMIŲ IŠSKYRIMO PUSPROGRAMĖ

```
## Importuojamos bibliotekos
import os
import numpy as np
import scipy.io.wavfile as spiowav
import scipy.signal as scsig
import glob
import MFCC_pozymiai
import gmm
import signalo_kadru_aktyvumas as evad
from collections import Counter

## Įvesties duomenys
FILES=glob.glob('russian records*\*.wav') # sudaromas pasirinkto aplanko garso
įrašų failų sąrašas (pakeisti pagal kalbą)
ubm_file = 'GMM.txt' # Nuskaitomas UBM failo vardas

## Nuskaitomi statistiniai duomenys:
fmean = np.load('fmean.npy') # komponentės vidurkiai
smean = np.load('smean.npy') # komponentės standartiniai nuokrypiai

# Sudaromi indeksų sąrašai
columns_idx = np.hstack( ([1,2], np.arange(4,60) ) ) # Sudaromas sąrašas iki 60,
neįterpiant indeksų, kurių reikšmės 0 ir 3
columns_rest_idx = np.arange(2,58) # Sudaromas sąrašas nuo 2 iki 58
# Randamos indeksus columns_idx atitinkančios statistinių duomenų fmean bei
smean reikšmės
fmean = fmean[columns_idx]
smean = smean[columns_idx]

## Konfigūracijos parametrai
SOURCERATE    = 1250 # Atskaitų periodas (HTK laiko vienetais)
TARGETRATE    = 100000 # Laiko intervalai (HTK laiko vienetais)
LOFREQ        = 120 # Žemųjų dažnių riba (Hz)
HIFREQ        = 3800 # Aukštųjų dažnių riba (Hz)
WINDOWSIZE    = 250000.0 # Lango ilgis (HTK laiko vienetais)
PREEMCOEF     = 0.97 # Pradinio filtravimo koeficientas
NUMCHANS      = 24 # Filtrų rinkinio filtrų skaičius (vnt.)
CEPLIFTER     = 22 # Kepstro filtrų skaičius (vnt.)
NUMCEPS       = 19 # Kepstro koeficientų skaičius
deltawindow   = accwindow = 2 # Skirtuminių požymių skaičiavimo langas
cmvn_lc       = 150 # Kadrų kiekis iš kairės pusės nuo slankiojančio lango esa-
mos padėties
cmvn_rc       = 150 # Kadrų kiekis iš dešinės pusės nuo slankiojančio lango esa-
mos padėties
fs            = 1e7/SOURCERATE # Imties dažnis (Hz)
window        = WINDOWSIZE/SOURCERATE # Kadro lango ilgis atskaitomis
noverlap      = (WINDOWSIZE-TARGETRATE)/SOURCERATE # Persidengimų tarp gretimų
kadrų ilgis atskaitomis
ESCALE        = 0.1 # logaritminės energijos mastelis
SILFLOOR      = 50.0 # Energijos apatinė slenkstinė riba (dB)

## Loginiai parametrai
ZMEANSOURCE   = True # Vidurkio atėmimas iš signalo
USEHAMMING    = True # Hammingo lango naudojimas
RAWENERGY     = True # Energijos požymio gražinimas
```

```
ENORMALISE    = True # Energijos normavimas

## Pagalbinės funkcijos
def load_ubm(fname):
    gmm = np.loadtxt(fname, dtype=np.float32)
    n_superdims = (gmm.shape[1] - 1) // 2
    weights = gmm[:,0]
    means   = gmm[:,1:(n_superdims+1)]
    covs    = gmm[:,(n_superdims+1):]
    return weights, means, covs

def compute_vad(s, win_length=160, win_overlap=80):
    v = evad.compute_vad(s,win_length=win_length, win_overlap=win_overlap, n_re-
alignment=10)
    n_frames = sum(v)
    n_regions = n_frames
    return v, n_regions, n_frames

## Parengiamasis duomenų apdorojimas
# kviečiama Mel filtrų rinkinio apskaičiavimo funkcija
fbank_mx      = MFCC_pozymiai.mel_fbank_mx(winlen_nfft = window,
                                           fs          = fs,
                                           NUMCHANS    = NUMCHANS,
                                           LOFREQ      = LOFREQ,
                                           HIFREQ      = HIFREQ)

## Nuskaitomi akustinio modelio parametrai
#(pakeisti pagal modelį: 78 eilutė - 1 modelis, 79-81 eilutės - 2 modelis,
# 80-82 eilutės - pasirinktos kalbos k-vidurkių modelis, 83-85 eilutės - pasi-
rinktos kalbos GMM modelis)
#ubm_weights, ubm_means, ubm_covs = load_ubm(ubm_file)
#ubm_means = np.load('kmeans_en.npy')
ubm_weights = np.ones( (2048,), dtype=np.float64)/2048
ubm_covs= np.ones( (2048,60), dtype=np.float64)
ubm_means = np.load('russian\\Models\\KMEAN\\test_ubm_19.npy') # (pakeisti pagal
kalbą)
#ubm_weights = np.load('russian\\Models\GMM\\w10.npy') # (pakeisti pagal kalbą)
#ubm_means = np.load('russian\\Models\GMM\\mu10.npy') # (pakeisti pagal kalbą)
#ubm_covs = np.load('russian\\Models\GMM\\sigma10.npy') # (pakeisti pagal kalbą)

# Vykdoma GMM modelio inicializavimo funkcija
GMM = gmm.gmm_eval_prep(ubm_weights, ubm_means, ubm_covs)

# Vykdomas UBM statistikų normavimas jei UBM vidurkių matricos ubm_menas stul-
pelių skaičius atitinka UBM kovariacijų matricos ubm_covs stulpelių skaičių
dimF=ubm_means.shape[1] # Randamas  UBM vidurkių matricos ubm_menas stulpelių
skaičius
if ubm_covs.shape[1] == dimF:
    ubm_norm = 1/np.sqrt(ubm_covs);

for records in FILES:
    ## Vykdomas įrašo nuskaitymas
    print('records = %s' %records) # Spausdinamas įrašo pavadinimas iš sąrašo
FILES
    rate,sigraw = spiowav.read(records) # Vykdomas įrašo nuskaitymas (dažnis,
signalas)

    ## Vykdoma įrašo duomenų konfigūracija
    # Grąžinamas pirmasis signalo sigraw stulpelis jei signalo dydis ne mažesnis
nei 1
    if len(sigraw.shape)>1:
```

```
        sigraw = sigraw[:,0]
    # Pakeičiamas signalo atskaitų dažnis į 8 kHz jei dažnis nelygus 8 kHz
    if rate != 8000:
        secs = len(sigraw)/rate
        samps = secs*8000
        sigraw = scsig.resample(sigraw, samps)
    sig = np.float64(sigraw);

    ## Randami ir spausdinami: dažnis, signalo ilgis, MFCC požymiai (kviečiama
MFCC požymių apskaičiavimo funkcija)
    print('rate ', rate)
    print('sig length ', len(sig))
    print('  Extracting features'),
    fea = MFCC_pozymiai.mfcc_htk(sig,
                window      = np.int32(window),
                noverlap    = np.int32(noverlap),
                fbank_mx    = fbank_mx,
                _0          = 'first',
                NUMCEPS     = NUMCEPS,
                RAWENERGY   = RAWENERGY,
                PREEMCOEF   = PREEMCOEF,
                CEPLIFTER   = CEPLIFTER,
                ZMEANSOURCE = ZMEANSOURCE,
                ENORMALISE  = ENORMALISE,
                ESCALE      = ESCALE,
                SILFLOOR    = SILFLOOR,
                USEHAMMING  = USEHAMMING)

    ## MFCC papildymas pirmos ir antros eilės skirtuminiais požymiais
    fea = MFCC_pozymiai.add_deriv(fea,(deltawindow,accwindow))
    print('  Reshaping to SFeaCat convention')
    ## MFCC požymiai išrikiuojami pagal SFEaCut
    fea = fea.reshape(fea.shape[0], 3, -1).trans-
pose((0,2,1)).reshape(fea.shape[0],-1)

    ## Vykdomas signalo aktyvaus kadro nustatymas
    print('  Computing VAD ')
    vad,n_regions,n_frames = compute_vad(sig, win_length=np.int32(window),
win_overlap=np.int32(noverlap)) [:len(fea)]     # Kviečiama aktyvaus kadro apti-
kimo funkcija

    ## Atliekamas požymių normavimas
    fea_03 = MFCC_pozymiai.cmvn_floating(fea[:,[0,3]], cmvn_lc, cmvn_rc, unbia-
sed=True) # atskiriami MFCC 0 ir 3 požymiai ir normuojami pagal failo duomenis -
kviečiama vidurkio ir dispersijos normavimo slankiojančio lango atžvilgiu funk-
cija
    fea_rest = fea[:,columns_idx] # naujai matricai priskiriami likę požymiai
(be 0 ir 3)
    fea2_rest = (fea_rest - fmean) / smean # likusios požymius normuoja pagal
statistinius duomenis
    fea2 = np.hstack( (fea_03[:,0][:,None], fea2_rest[:,[0,1]],
fea_03[:,1][:,None], fea2_rest[:,columns_rest_idx]) ) # požymiai sustatomi anks-
tesniąja tvarka

    ## Randami parametrai: GMM tikėtinumo reikšmės, indeksai bei LLK
    data_sqr = fea2[:, GMM['utr']] * fea2[:, GMM['utc']]  # vykdomas kvadratinis
MFCC požymių išplėtimas
    gamma = -0.5 * data_sqr.dot(GMM['invCovs'].T) + fea2.dot(GMM['in-
vCovMeans'].T) + GMM['gconsts'] # randamos GMM komponenčių tikėtinumo reikšmės
    idx = np.int32(np.nanargmax(gamma, axis =1)) # surandami indeksai GMM kompo-
nenčių su maksimaliomis tikėtinumo reikšmėmis
```

```
    LLK = np.int32(np.nanmax(gamma, axis =1)) # randamos indeksus atitinkančios
GMM tikėtinumo reikšmės

    ## Atskiriamos balso sričių komponentės „voice" nuo fono sričių komponenčių
„background" taikant „kaukių metodą"
    print('  Applying VAD [#frames=' + repr(n_frames) + ', #regions=' +
repr(n_regions) + ']') # spausdinama, kad taikomi po aktyvaus kadro signale ap-
tikimo gauti rezultatai
    mask_power = fea[:,0]>30 # 1)          sukuriama loginė matrica mask_power,
kuri teisinga, kai požymių matricos pirmo stulpelio elementai didesni už 30
    mask = (vad==0) * mask_power # randami balso sričių indeksai - sukuriama lo-
ginė matrica mask, kuri teisinga, kai signalo kadras aktyvus ir loginė matrica
mask_power teisingi
    voice = idx[mask] # 3)       nustatomi GMM balso sričių indeksai - pagal lo-
ginės matricos mask teisingas reikšmes surašomi balso sričių indeksai
    mask1 = np.invert(mask) # nustatomi fono sričių indeksai - randama atvirkš-
tinė matrica (mask)^(-1)
    background = idx[mask1] # nustatomi fono sričių GMM indeksai - pagal at-
virkštinės matricos (mask)^(-1) teisingas reikšmes surašomi fono sričių indeksai
    fea2 = fea2[mask,:] # nustatomi balso sričių požymiai

    ## Randamas komponenčių pasikartojimo dažnumas
    voice_times = Counter(voice) #išskiriama ir suskaičiuojama kiek kartų pasi-
kartoja balso sričių komponentės
    background_times = Counter(background) #išskiriama ir suskaičiuojama kiek
kartų pasikartoja fono sričių komponentės

    ## Rezultatų išsaugojimas
    # Nuskaitomas failo pavadinimas
    fpath, fname = os.path.split(records)
    label=fname.split('.')[0]
    np.save('italian\\features\\' + label +' fea', fea2) # išsaugomi balso sri-
čių požymiai (pakeisti pagal kalbą)
    # jei įraše kalbėjo moteris - išsaugomi balso arba fono sričių komponenčių
pasikartojimo dažniai moterų įrašų balso (voice_f) ar fono (background_f) ap-
lanke atitinkamai
    if "-f-"  in label:
        np.save('russian\\Statistics\\KMEAN\\voice_f\\' + label +' voice times',
sorted(voice_times.items())) # (pakeisti pagal kalbą ir modelį)
        np.save('russian\\Statistics\\KMEAN\\background_f\\' + label +' bac-
kground times', sorted(background_times.items())) # (pakeisti pagal kalbą ir mo-
delį)

    # jei įraše kalbėjo vyras - išsaugomi balso arba fono sričių komponenčių pa-
sikartojimo dažnumai vyrų įrašų aplanke balso (voice_m) ar fono (background_m)
aplanke atitinkamai
    if "-m-"  in label:
        np.save('russian\\Statistics\\KMEAN\\voice_m\\' + label +' voice times',
sorted(voice_times.items())) # (pakeisti pagal kalbą ir modelį)
        np.save('russian\\Statistics\\KMEAN\\background_m\\' + label +' bac-
kground times', sorted(background_times.items())) # (pakeisti pagal kalbą ir mo-
delį)
```

STATISTIKŲ SKAIČIAVIMO PUSPROGRAMĖ

```
## Importuojamos bibliotekos
import numpy as np
import glob

## Įvesties duomenys: modelių komponenčių sąrašai
FILES_voice_f = glob.glob('russian\\Statistics\\KMEAN\\voice_f*\*.npy') # (pakeisti pagal kalbą)
FILES_voice_m = glob.glob('russian\\Statistics\\KMEAN\\voice_m*\*.npy') # (pakeisti pagal kalbą)
FILES_bg_f    = glob.glob('russian\\Statistics\\KMEAN\\background_f*\*.npy') # (pakeisti pagal kalbą)
FILES_bg_m    = glob.glob('russian\\Statistics\\KMEAN\\background_m*\*.npy') # (pakeisti pagal kalbą)

## Parametrai: sukuriamos tuščios arba nulinės matricos tolimesniam duomenų apdorojimui
voice_f_len   = np.zeros( (len(FILES_voice_f),), dtype=np.int32) # moterų balso sričių komponenčių ilgiai įrašuose
voice_m_len   = np.zeros( (len(FILES_voice_m),), dtype=np.int32) # vyrų balso sričių komponenčių ilgiai įrašuose
bg_f_len      = np.zeros( (len(FILES_bg_f),), dtype=np.int32) # moterų fono sričių komponenčių ilgiai įrašuose
bg_m_len      = np.zeros( (len(FILES_bg_m),), dtype=np.int32) # vyrų fono sričių komponenčių ilgiai įrašuose
voice_f       = np.zeros( (2048,), dtype=np.int32)  # moterų balso sričių komponenčių pasikartojimai įrašuose
voice_m       = np.zeros( (2048,), dtype=np.int32)  # vyrų balso sričių komponenčių  pasikartojimai įrašuose
bg_f          = np.zeros( (2048,), dtype=np.int32)  # moterų fono sričių komponenčių  pasikartojimai įrašuose
bg_m          = np.zeros( (2048,), dtype=np.int32)  # vyrų fono sričių komponenčių pasikartojimai įrašuose
voice_times_f = [] # moterų balso sričių komponenčių sugrupuoti pasikartojimai įrašuose
voice_times_m = [] # vyrų balso sričių komponenčių sugrupuoti pasikartojimai įrašuose
bg_times_f    = [] # moterų fono sričių komponenčių sugrupuoti pasikartojimai įrašuose
bg_times_m    = [] # vyrų fono sričių komponenčių sugrupuoti pasikartojimai įrašuose
voice_f_idx   = []; # moterų balso sričių komponenčių indeksų sąrašas
voice_m_idx   = []; # vyrų balso sričių komponenčių indeksų sąrašas
bg_f_idx      = []; # moterų fono sričių komponenčių indeksų sąrašas
bg_m_idx      = []; # vyrų fono sričių komponenčių indeksų sąrašas

## Tikrinama, ar yra tuščių požymių masyvų, jei taip - priskiriama vertė 0
for j,vf_qual in enumerate(FILES_voice_f): # tikrinamos moterų balso sričių komponentėms
     print('Failas, tuščių paieška moterų balso sričių komponentėms %d' %(j))
     vf = np.load(vf_qual)
     if len(vf)==0:
         vf = np.zeros((1,2), dtype=np.int)
         np.save(vf_qual,vf)
for j,vm_qual in enumerate(FILES_voice_m): # tikrinamos vyrų balso sričių komponentėms
```

```
        print('Failas, tuščių paieška vyrų balso sričių komponentėms %d' %(j))
        vm = np.load(vm_qual)
        if len(vm)==0:
            vm = np.zeros((1,2), dtype=np.int)
            np.save(vm_qual,vm)
for j,bgf_qual in enumerate(FILES_bg_f): # tikrinamos moterų fono sričių kompo-
nentėms
        print('Failas, tuščių paieška moterų fono sričių komponentėms %d' %(j))
        bgf = np.load(bgf_qual)
        if len(bgf)==0:
            bgf = np.zeros((1,2), dtype=np.int)
            np.save(bgf_qual,bgf)
for j,bgm_qual in enumerate(FILES_bg_m): # tikrinamos vyrų fono sričių komponen-
tėms
        print('Failas, tuščių paieška vyrų fono sričių komponentėms %d' %(j))
        bgm = np.load(bgm_qual)
        if len(bgm)==0:
            bgm = np.zeros((1,2), dtype=np.int)
            np.save(bgm_qual,bgm)

## Balso sričių komponenčių statistikų ir pasikartojimų įrašuose nustatymas
# Moterų įrašuose
for j,v_qual in enumerate(FILES_voice_f):
    print('Failas moterų balso sričių komponentės %d' %(j))
    v = np.load(v_qual)
    voice_f_len[j] = np.sum(v[:,1], axis=0)
    for i in range(len(v)):
        if v[i][1]>1:
            voice_f[v[i][0]] = voice_f[v[i][0]] + v[i][1]
for i in range (0,len(voice_f)):
    if voice_f[i]>0:
        voice_times_f.append([i,voice_f[i]])
# Vyrų įrašuose
for j,v_qual in enumerate(FILES_voice_m):
    print('Failas vyrų balso sričių komponentės %d' %(j))
    v = np.load(v_qual)
    voice_m_len[j] = np.sum(v[:,1], axis=0)
    for i in range(len(v)):
        if v[i][1]>1:
            voice_m[v[i][0]] = voice_m[v[i][0]] + v[i][1]
for i in range (0,len(voice_m)):
    if voice_m[i]>0:
        voice_times_m.append([i,voice_m[i]])

## Fono sričių komponenčių statistikų ir pasikartojimų įrašuose skaičiavimas
# Moterų įrašuose
for j,bg_qual in enumerate(FILES_bg_f):
    print('Failas moterų fono sričių komponentės %d' %(j))
    bg = np.load(bg_qual)
    bg_f_len[j] = np.sum(bg[:,1], axis=0)
    for i in range(len(bg)):
        if bg[i][1]>1:
            bg_f[bg[i][0]] = bg_f[bg[i][0]] + bg[i][1]
for i in range (0,len(bg_f)):
    if bg_f[i]>0:
        bg_times_f.append([i,bg_f[i]])

# Vyrų įrašuose
for j,bg_qual in enumerate(FILES_bg_m):
    print('Failas vyrų fono sričių komponentės %d' %(j))
    bg = np.load(bg_qual)
```

```
    bg_m_len[j] = np.sum(bg[:,1], axis=0)
    for i in range(len(bg)):
        if bg[i][1]>1:
            bg_m[bg[i][0]] = bg_m[bg[i][0]] + bg[i][1]
for i in range (0,len(bg_m)):
    if bg_m[i]>0:
        bg_times_m.append([i,bg_m[i]])

##  Randamos balso ir fono sričių užimamų komponenčių procentinės išraiškos
# Sudaromi komponenčių indeksų sąrašai
for i in range(len(voice_times_f)): # Moterų balso sričių komponentėms
    voice_f_idx.append(voice_times_f[i][0])
for i in range(len(voice_times_m)): # Vyrų balso sričių komponentėms
    voice_m_idx.append(voice_times_m[i][0])
for i in range(len(bg_times_f)): # Moterų fono sričių komponentėms
    bg_f_idx.append(bg_times_f[i][0])
for i in range(len(bg_times_m)): # Vyrų fono sričių komponentėms
    bg_m_idx.append(bg_times_m[i][0])

# Apskaičiuojamos užimamų komponenčių procentinės išraiškos
VOICE_f = np.zeros( (len(voice_times_f),len(FILES_voice_f)), dtype=np.float32) #
moterų balso sričių komponenčių procentinės išraiškos
for j,g_qual in enumerate(FILES_voice_f): # Moterų balso sričių komponentėms
     print('Failas balsas moteris (procent) %d' %(j))
     voice_times_f = np.load(g_qual)
     for m in range(len(voice_f_idx)):
        if voice_f_idx[m] in voice_times_f[:,0]:
            item_index = np.where(voice_times_f[:,0]==voice_f_idx[m])
            VOICE_f[m,j]=100*voice_times_f[item_index[0][0]][1]/voice_f_len[j]
VOICE_m = np.zeros( (len(voice_times_m),len(FILES_voice_m)), dtype=np.float32) #
vyrų balso sričių komponenčių procentinės išraiškos
for j,g_qual in enumerate(FILES_voice_m): # Vyrų balso sričių komponentėms
     print('Failas balsas vyras (procent) %d' %(j))
     voice_times_m = np.load(g_qual)
     for m in range(len(voice_m_idx)):
        if voice_m_idx[m] in voice_times_m[:,0]:
            item_index = np.where(voice_times_m[:,0]==voice_m_idx[m])
            VOICE_m[m,j]=100*voice_times_m[item_index[0][0]][1]/voice_m_len[j]
BG_f = np.zeros( (len(bg_times_f),len(FILES_bg_f)), dtype=np.float32) # moterų
fono sričių komponenčių procentinės išraiškos
for j,p_qual in enumerate(FILES_bg_f): # Moterų fono sričių komponentėms
     print('Failas fonas moteris (procent) %d' %(j))
     bg_times_f = np.load(p_qual)
     for m in range(len(bg_f_idx)):
        if bg_f_idx[m] in bg_times_f[:,0]:
            item_index = np.where(bg_times_f[:,0]==bg_f_idx[m])
            BG_f[m,j]=100*bg_times_f[item_index[0][0]][1]/bg_f_len[j]
BG_m = np.zeros( (len(bg_times_m),len(FILES_bg_m)), dtype=np.float32) # vyrų
fono sričių komponenčių procentinės išraiškos
for j,p_qual in enumerate(FILES_bg_m): # Vyrų fono sričių komponentėms
     print('Failas fonas vyras (procent) %d' %(j))
     bg_times_m = np.load(p_qual)
     for m in range(len(bg_m_idx)):
        if bg_m_idx[m] in bg_times_m[:,0]:
            item_index = np.where(bg_times_m[:,0]==bg_m_idx[m])
            BG_m[m,j]=100*bg_times_m[item_index[0][0]][1]/bg_m_len[j]

## Balso ir fono sričių užimamų komponenčių procentinių išraiškų kvartilių skai-
čiavimas
# Sukuriami pagalbiniai parametrai
```

```
voice_f_q     =  np.zeros( (len(VOICE_f),6), dtype=np.float32) # Moterų balso sričių komponenčių kvartilių parametras
voice_m_q     =  np.zeros( (len(VOICE_m),6), dtype=np.float32) # Vyrų balso sričių komponenčių kvartilių parametras
bg_f_q        =  np.zeros( (len(BG_f),6), dtype=np.float32) # Moterų fono sričių komponentčių kvartilių parametras
bg_m_q        =  np.zeros( (len(BG_m),6), dtype=np.float32) # Vyrų fono sričių komponentčių kvartilių parametras

# Moterų balso sričių komponentėms
for i in range (len(VOICE_f)):
    voice_f_q[i,0] = np.percentile(VOICE_f[i,:], 0) # Q = 0
    voice_f_q[i,1] = np.percentile(VOICE_f[i,:], 25) # Q = 0,25
    voice_f_q[i,2] = np.percentile(VOICE_f[i,:], 50) # Q = 0,50
    voice_f_q[i,3] = np.percentile(VOICE_f[i,:], 75) # Q = 0,75
    voice_f_q[i,4] = np.percentile(VOICE_f[i,:], 100) # Q = 1
    voice_f_q[i,5] = voice_f_idx[i] # priskiriama komponentės reikšmė
# Vyrų balso sričių komponentėms
for i in range (len(VOICE_m)):
    voice_m_q[i,0] = np.percentile(VOICE_m[i,:], 0) # Q = 0
    voice_m_q[i,1] = np.percentile(VOICE_m[i,:], 25) # Q = 0,25
    voice_m_q[i,2] = np.percentile(VOICE_m[i,:], 50) # Q = 0,50
    voice_m_q[i,3] = np.percentile(VOICE_m[i,:], 75) # Q = 0,75
    voice_m_q[i,4] = np.percentile(VOICE_m[i,:], 100) # Q = 1
    voice_m_q[i,5] = voice_m_idx[i] # priskiriama komponentės reikšmė
# Moterų fono sričių komponentėms
for i in range (len(BG_f)):
    bg_f_q[i,0] = np.percentile(BG_f[i,:], 0) # Q = 0
    bg_f_q[i,1] = np.percentile(BG_f[i,:], 25) # Q = 0,25
    bg_f_q[i,2] = np.percentile(BG_f[i,:], 50) # Q = 0,50
    bg_f_q[i,3] = np.percentile(BG_f[i,:], 75) # Q = 0,75
    bg_f_q[i,4] = np.percentile(BG_f[i,:], 100) # Q = 1
    bg_f_q[i,5] = bg_f_idx[i] # priskiriama komponentės reikšmė
 # Vyrų fono sričių komponentėms
for i in range (len(BG_m)):
    bg_m_q[i,0] = np.percentile(BG_m[i,:], 0) # Q = 0
    bg_m_q[i,1] = np.percentile(BG_m[i,:], 25) # Q = 0,25
    bg_m_q[i,2] = np.percentile(BG_m[i,:], 50) # Q = 0,50
    bg_m_q[i,3] = np.percentile(BG_m[i,:], 75) # Q = 0,75
    bg_m_q[i,4] = np.percentile(BG_m[i,:], 100) # Q = 1
    bg_m_q[i,5] = bg_m_idx[i] # priskiriama komponentės reikšmė

## Gautų kvartilių išrikiavimas pagal medianą
voice_f_q_median = voice_f_q[np.argsort(voice_f_q[:,2])] # Moterų balso sričių komponentėms
voice_m_q_median = voice_m_q[np.argsort(voice_m_q[:,2])] # Vyrų balso sričių komponentėms
bg_m_f_median    = bg_f_q[np.argsort(bg_f_q[:,2])] # Moterų fono sričių komponentėms
bg_m_q_median    = bg_m_q[np.argsort(bg_m_q[:,2])] # Vyrų fono sričių komponentėms

## Rezultatų išsaugojimas
np.save('russian\\Statistics\\KMEAN\\voice times f.npy', voice_times_f) #(pakeisti pagal kalbą)
np.save('russian\\Statistics\\KMEAN\\voice times m.npy', voice_times_m) #(pakeisti pagal kalbą)
np.save('russian\\Statistics\\KMEAN\\background times f.npy', bg_times_f) #(pakeisti pagal kalbą)
np.save('russian\\Statistics\\KMEAN\\background times m.npy', bg_times_m) #(pakeisti pagal kalbą)
```

```
np.save('russian\\Statistics\\KMEAN\\voice len f.npy', voice_f_len) #(pakeisti
pagal kalbą)
np.save('russian\\Statistics\\KMEAN\\voice len m.npy', voice_m_len) #(pakeisti
pagal kalbą)
np.save('russian\\Statistics\\KMEAN\\background len f.npy', bg_f_len) #(pakeisti
pagal kalbą)
np.save('russian\\Statistics\\KMEAN\\background len m.npy', bg_m_len) #(pakeisti
pagal kalbą)
np.save('russian\\Statistics\\KMEAN\\Voice female.npy', VOICE_f) #(pakeisti pa-
gal kalbą)
np.save('russian\\Statistics\\KMEAN\\Voice male.npy', VOICE_m) #(pakeisti pagal
kalbą)
np.save('russian\\Statistics\\KMEAN\\Background female.npy', BG_f) #(pakeisti
pagal kalbą)
np.save('russian\\Statistics\\KMEAN\\Background male.npy', BG_m) #(pakeisti pa-
gal kalbą)
np.save('russian\\Statistics\\KMEAN\\Voice sorted percentile female.npy',
voice_f_q_median) #(pakeisti pagal kalbą)
np.save('russian\\Statistics\\KMEAN\\Voice sorted percentile male.npy',
voice_m_q_median) #(pakeisti pagal kalbą)
np.save('russian\\Statistics\\KMEAN\\Background sorted percentile female.npy',
bg_m_f_median) #(pakeisti pagal kalbą)
np.save('russian\\Statistics\\KMEAN\\Background sorted percentile male.npy',
bg_m_q_median) #(pakeisti pagal kalbą)
```

SIGNALO AKTYVIŲ KADRŲ NUSTATYMO PUSPROGRAMĖ

```
## Importuojamos bibliotekos
import gmm
import numpy as np

## Pagalbinė funkcija
def framing(a, window, shift=1):
    shape = ((a.shape[0] - window) / shift + 1, window) + a.shape[1:]
    strides = (a.strides[0]*shift,a.strides[0]) + a.strides[1:]
    return np.lib.stride_tricks.as_strided(a, shape=shape, strides=strides)

## Pagrindinė funkcija
def compute_vad(s, win_length=160, win_overlap=80, n_realignment=5,
threshold=0.3):
    s = s**2
    F = framing(s, win_length, win_length - win_overlap)
    E = F.sum(axis=1)
    E -= E.mean()
    E /= E.std()
    mm = np.array((-1.00, 0.00, 1.00))[:, np.newaxis]
    ee = np.array(( 1.00, 1.00, 1.00))[:, np.newaxis]
    ww = np.array(( 0.33, 0.33, 0.33))
    GMM = gmm.gmm_eval_prep(ww, mm, ee)
    E = E[:,np.newaxis]
    llh, N, F, S = gmm.gmm_eval(E, GMM, return_accums=2)
    ww, mm, ee   = gmm.gmm_update(N, F, S)
    GMM = gmm.gmm_eval_prep(ww, mm, ee)
    llhs = gmm.gmm_llhs(E, GMM)
    llh  = gmm.logsumexp(llhs, axis=1)[:,np.newaxis]
    llhs = np.exp(llhs - llh)
    out  = np.zeros(llhs.shape[0], dtype=np.bool)
    out[llhs[:,0] < threshold] = True
    return out
```

AKUSTINIO MODELIO SUDARYMO TAIKANT GMM PUSPROGRAMĖ

```
## Importuojamos bibliotekos
import fnmatch
import numpy as np
import os

## Įvesties duomenys
FeaDir = 'russian\\features\\' # (pakeisti pagal kalbą)
FILES = fnmatch.filter(os.listdir(FeaDir), '*.npy')

## Konfigūracijos parametrai
nFiles = len(FILES) # komponenčių skaičius įrašuose
nFeatures = 60 # inicijuojamas požymių kiekis
mixList = [1,  2,  4,  8,  16,  32,  64,  128, 256, 512, 1024,  2048] # mišinio
komponenčių sąrašas
niter = [1, 2, 4, 4,  4,  4,  6,  6,   10,  10,   15,   15] # iteracijų sąrašas
var_const = 0.1 # kovariacijų slenkstinė riba

# Sukuriamos GMM parametrų tuščia, nulinė bei vienetinės matricos tolimesniam
duomenų apdorojimui
gmm={} # GMM matrica
gmm['mu'] = np.zeros( (nFeatures,1), dtype=np.float64) # GMM vidurkių vektorius
gmm['sigma'] = np.ones( (nFeatures,1), dtype=np.float64)# GMM kovariacinės mat-
ricos
gmm['w'] = np.ones( (1,1), dtype=np.float64) # GMM svoriai


## Pagalbinės funkcijos
def logsumexp(x):
    xmax = np.max(x, axis=0);
    y    = xmax + np.log(np.sum(np.exp(x-xmax), axis=0))
    ind  = np.where(np.logical_not(np.isfinite(xmax)))
    if len(ind)>0:
        y[ind] = xmax[ind]
    return y

def lgmmprob(data, mu, sigma, w):
    ndim = len(data)
    C = np.sum((mu * (mu/sigma)), axis=0) + np.sum(np.log(sigma),axis=0)
    D = np.dot((1/sigma).T,data*data) - 2*np.dot((mu/sigma).T,data) +
ndim*np.log(2*np.pi)
    logprob = -0.5 * (C[:,np.newaxis] + D)
    if len(w)>1:
        wnew = np.copy(w)
        wnew = wnew[:,np.newaxis]
    else:
        wnew = np.copy(w)
    logprob = logprob +  np.log(wnew)
    return logprob

def postprob(data, mu, sigma, w):
    post = lgmmprob(data, mu, sigma, w)
    llk  = logsumexp(post);
    post = np.exp(post - llk);
    return post, llk
```

```
def apply_var_floors(w, sigma, floor_const):
    vFloor = np.dot(sigma,w.T) * floor_const
    vFloor = vFloor[:,np.newaxis]
    sigma  = np.maximum(sigma, vFloor)
    return sigma

def expectation(data, gmm):
    post, llk = postprob(data, gmm['mu'], gmm['sigma'], gmm['w'])
    N = np.sum(post, axis=1).T
    F = np.dot(data, post.T)
    S = np.dot((data*data), post.T)
    return N, F, S, llk

def maximization(N, F, S):
    w  = N / np.sum(N)
    mu = F / N
    sigma = S / N - (mu * mu)
    sigma = apply_var_floors(w, sigma, var_const)
    gmm['w'] = np.copy(w)
    gmm['mu'] = np.copy(mu)
    gmm['sigma'] = np.copy(sigma)
    return gmm

def gmm_mixup(gmm, nmix):
    mu = np.copy(gmm['mu'])
    sigma = np.copy(gmm['sigma'])
    w = np.copy(gmm['w'])
    ndim, nmix = sigma.shape
    idx = np.int32(np.nanargmax(sigma, axis=0))
    eps = np.sqrt(np.int32(np.nanmax(sigma, axis=0)))
    munew = np.zeros( (60,0), dtype=np.float64)
    for i in range(nmix):
        mu_modif1 = np.copy(mu[:,i])
        mu_modif1[idx[i]] -= eps[i]
        mu_modif1 = mu_modif1[:,np.newaxis]
        munew = np.hstack((munew,mu_modif1))
    for i in range(nmix):
        mu_modif2 = np.copy(mu[:,i])
        mu_modif2[idx[i]] += eps[i]
        mu_modif2 = mu_modif2[:,np.newaxis]
        munew = np.hstack((munew,mu_modif2))
    mu = np.copy(munew)
    sigma = np.hstack((sigma,sigma))
    w = np.hstack((w, w))*0.5
    gmm['w']  = w
    gmm['mu'] = mu
    gmm['sigma'] = sigma
    return gmm


# Vykdomas 12 iteracijų EM algoritmas
for i in range(11):
    mix = mixList[i]
    print('\nRe-estimating the GMM hyperparameters for %d components ...\n'
%mix)
    for ii in range(niter[i]):
        print('EM iter#: %d \t' %ii)
        N = 0.0
        F = np.zeros( (nFeatures,mixList[i]), dtype=np.float64)
        S = np.zeros( (nFeatures,mixList[i]), dtype=np.float64)
        L = 0.0
```

```
        nframes = 0
        for j,file in enumerate(FILES):
            if np.mod(j,1000)==0:
                print('file: %d \t' %j)
            fea = np.load(FeaDir + file)
            fea = fea.T
            n, f, s, l = expectation(fea, gmm)
            N = N + n
            F = F + f
            S = S + s
            L = L + np.sum(l)
            nframes = nframes + len(l)

        print('[llk = %.2f] \t [elaps = %.2f s]\n' %(L/nframes, 0) )
        gmm = maximization(N, F, S)

    np.save('russian\\GMM\\mu'+str(i)+'.npy', gmm['mu']) # (pakeisti pagal
kalbą)
    np.save('russian\\GMM\\sigma'+str(i)+'.npy', gmm['sigma'] ) # (pakeisti pa-
gal kalbą)
    np.save('russian\\GMM\\w'+str(i)+'.npy', gmm['w']) # (pakeisti pagal kalbą)
    gmm = gmm_mixup(gmm,mix)
```

KALBOS SIGNALŲ AKUSTINIO MODELIO SUDARYTO K-VIDURKIŲ KLASTERIZAVIMO METODU PUSPROGRAMĖ

```
## Importuojamos bibliotekos
import fnmatch
import numpy as np
import os

## Įvesties parametrai (požymiai)
FeaDir = 'russian\\features\\' # (pakeisti pagal kalbą)
Files = fnmatch.filter(os.listdir(FeaDir), '*.npy')
nFiles = len(Files)

## Konfigūracijos parametrai
dimensions = 60 # požymių skaičius
num_clusters = 2048  # modelio komponenčių skaičius
# atsitiktinai sugeneruojami centroidų centrai
centroids = np.random.normal(0,1, num_clusters*dimensions)
centroids = centroids.reshape( (num_clusters,dimensions))
idx_clusters = np.ones( (num_clusters,), dtype=np.int32)

## vykdomas k-vidurkių klasterizavimo algoritmas
for I in range(20):
    print(I)
    counts = np.zeros( (num_clusters,), dtype = np.int32)
    accum_sums = np.zeros( (num_clusters,dimensions), dtype = np.float32)
    distances = np.zeros( (num_clusters,), dtype = np.float32)
    for i in range(len(Files)):
        if i % 100 == 0:
            print('i=%d' %i)
        fea = np.load(FeaDir + Files[i])
        for j in range(len(fea)):
            ff = fea[idx_clusters*j,:]
            distances = np.sum((ff-centroids)**2, axis=1)
            idx = np.nanargmin(distances)
            accum_sums[idx] += fea[j]
            counts[idx] += 1
    centroids = accum_sums / counts[:,None]
    np.save('russian\\Models\\KMEAN\\test_ubm_' + str(I) + '.npy', centroids) #
(pakeisti pagal kalbą)
    np.save('russian\\Models\\KMEAN\\test_ubm_counts_' + str(I) + '.npy', co-
unts) # (pakeisti pagal kalbą)
    mask = counts==0
    nnew = sum(mask)
    new_centroids = np.random.normal(0,1, nnew*dimensions)
    new_centroids = new_centroids.reshape( (nnew,dimensions))
                                                centroids[mask] = new_centroids
```

DAŽNIAUSIAI PASITAIKANČIŲ KOMPONENČIŲ NUSTATYTMO PUSPROGRAMĖ

```
## Importuojama biblioteka
import numpy as np

## Įvesties duomenys
# užimamų komponenčių procentinės išraiškos
VOICE_f          = np.load('spanish\\Statistics\\KMEAN\\Voice female.npy') #
(pakeisti pagal kalbą)
VOICE_m          = np.load('spanish\\Statistics\\KMEAN\\Voice male.npy') #(pa-
keisti pagal kalbą)
BG_f             = np.load('spanish\\Statistics\\KMEAN\\Background female.npy')
#(pakeisti pagal kalbą)
BG_m             = np.load('spanish\\Statistics\\KMEAN\\Background male.npy')
#(pakeisti pagal kalbą)
# komponenčių  pasikartojimai įrašuose
voice_times_f    = np.load('spanish\\Statistics\\KMEAN\\voice times f.npy')
#(pakeisti pagal kalbą)
voice_times_m    = np.load('spanish\\Statistics\\KMEAN\\voice times m.npy')
#(pakeisti pagal kalbą)
bg_times_f       = np.load('spanish\\Statistics\\KMEAN\\background times f.npy')
#(pakeisti pagal kalbą)
bg_times_m       = np.load('spanish\\Statistics\\KMEAN\\background times m.npy')
#(pakeisti pagal kalbą)
# užimamų komponenčių procentinių išraiškų kvartiliai išrikiuoti medianos atž-
vilgiu
voice_f_q_median = np.load('spanish\\Statistics\\KMEAN\\Voice sorted percentile
female.npy') #(pakeisti pagal kalbą)
voice_m_q_median = np.load('spanish\\Statistics\\KMEAN\\Voice sorted percentile
male.npy') #(pakeisti pagal kalbą)
bg_f_q_median    = np.load('spanish\\Statistics\\KMEAN\\Background sorted per-
centile female.npy') #(pakeisti pagal kalbą)
bg_m_q_median    = np.load('spanish\\Statistics\\KMEAN\\Background sorted per-
centile male.npy') #(pakeisti pagal kalbą)

# Inicializuojami dažniausiai pasitaikančių komponenčių procentinių išraiškų pa-
rametrai
voice_times_f_percent = []
voice_times_m_percent = []
bg_times_f_percent    = []
bg_times_m_percent    = []

#  komponenčių  pasikartojimai įrašuose išrikiuojami nuo rečiausiai  iki daž-
niausiai pasikartojančių
vf    = voice_times_f[np.argsort(voice_times_f[:,1])] # Moterų balso sričių kom-
ponentėms
vm    = voice_times_m[np.argsort(voice_times_m[:,1])] # Vyrų balso sričių kompo-
nentėms
bf    = bg_times_f[np.argsort(bg_times_f[:,1])] # Moterų fono sričių komponen-
tėms
bm    = bg_times_m[np.argsort(bg_times_m[:,1])] # Vyrų fono sričių komponentėms

## Apskaičiuojamos 6 dažniausiai pasikartojančių komponenčių užimamų sričių pro-
centinės išraiškos
# Moterų balso sričių
percent = 100*np.sum(vf[-6:5000,1])/np.sum(vf[:,1])
voice_times_f_percent = np.append(voice_times_f_percent,vf[-6:5000,0])
```

```
voice_times_f_percent = np.append(voice_times_f_percent,percent)
# Vyrų balso sričių
percent = 100*np.sum(vm[-6:5000,1])/np.sum(vm[:,1])
voice_times_m_percent = np.append(voice_times_m_percent,vm[-6:5000,0])
voice_times_m_percent = np.append(voice_times_m_percent,percent)
#  Moterų fono sričių
percent = 100*np.sum(bf[-6:5000,1])/np.sum(bf[:,1])
bg_times_f_percent = np.append(bg_times_f_percent,bf[-6:5000,0])
bg_times_f_percent = np.append(bg_times_f_percent,percent)
#  Vyrų fono sričių
percent = 100*np.sum(bm[-6:5000,1])/np.sum(bm[:,1])
bg_times_m_percent = np.append(bg_times_m_percent,bm[-6:5000,0])
bg_times_m_percent = np.append(bg_times_m_percent, percent)

# Rezultatų išsaugojimas
np.save('spanish\\Statistics\\KMEAN\\voice_times_f_percent',voice_times_f_per-
cent) #(pakeisti pagal kalbą)
np.save('spanish\\Statistics\\KMEAN\\voice_times_m_percent',voice_times_m_per-
cent) #(pakeisti pagal kalbą)
np.save('spanish\\Statistics\\KMEAN\\bg_times_f_percent', bg_times_f_percent)
#(pakeisti pagal kalbą)
np.save('spanish\\Statistics\\KMEAN\\bg_times_m_percent',bg_times_m_percent)
#(pakeisti pagal kalbą)
```

LOGARITMINIŲ TIKĖTINUMŲ APSKAIČIAVIMO PUSPROGRAMĖ

```
## Importuojamos bibliotekos
import numpy as np
import scipy.io.wavfile as spiowav
import scipy.signal as scsig
import glob
import MFCC_pozymiai
import gmm
import signalo_kadru_aktyvumas as evad
import scipy as sp

## Įvesties duomenys
FILES         = glob.glob('testing records\\german*\*.wav') # sudaromas pasi-
rinkto aplanko garso įrašų failų sąrašas (pakeisti pagal kalbą)
ubm_file      = 'GMM.txt' # nuskaitomas 1 modelio vardas
kmeans_en     = np.load('kmeans_en.npy') # nuskaitomas 2 modelio vardas
ubm_weights   = np.load('german\\Models\\GMM\\w10.npy') # nuskaitomi pasirinktos
kalbos GMM modelio svorių koeficientai (pakeisti pagal kalbą)
ubm_means     = np.load('german\\Models\\GMM\\mu10.npy') # nuskaitomas pasirink-
tos kalbos GMM modelio vidurkių vektorius (pakeisti pagal kalbą)
ubm_covs      = np.load('german\\Models\\GMM\\sigma10.npy') # nuskaitoma pasi-
rinktos kalbos GMM modelio kovariacijų matrica (pakeisti pagal kalbą)
kmean_means   = np.load('german\\Models\\KMEAN\\test_ubm_19.npy') # nuskaitomas
pasirinktos kalbos k-vidurkių modelio vidurkių vektorius(pakeisti pagal kalbą)
kmean_weights = np.ones((2048,), dtype=np.float64)/2048
kmean_covs    = np.ones((2048,60), dtype=np.float64)

## Nuskaitomi statistiniai duomenys:
fmean = np.load('fmean.npy') # komponentės vidurkiai
smean = np.load('smean.npy') # komponentės standartiniai nuokrypiai
# Sudaromi indeksų sąrašai
columns_idx      = np.hstack( ([1,2], np.arange(4,60) ) ) # Sudaromas sąrašas
iki 60, neįterpiant indeksų, kurių reikšmės 0 ir 3
columns_rest_idx = np.arange(2,58) # Sudaromas sąrašas nuo 2 iki 58
# Randamos indeksus columns_idx atitinkančios statistinių duomenų fmean bei
smean reikšmės
fmean = fmean[columns_idx]
smean = smean[columns_idx]

## Konfigūracijos parametrai
SOURCERATE    = 1250 # Atskaitų periodas (HTK laiko vienetais)
TARGETRATE    = 100000 # Laiko intervalai (HTK laiko vienetais)
LOFREQ        = 120 # Žemųjų dažnių riba (Hz)
HIFREQ        = 3800 # Aukštųjų dažnių riba (Hz)
WINDOWSIZE    = 250000.0 # Lango ilgis (HTK laiko vienetais)
PREEMCOEF     = 0.97 # Pradinio filtravimo koeficientas
NUMCHANS      = 24 # Filtrų rinkinio filtrų skaičius (vnt.)
CEPLIFTER     = 22 # Kepstro filtrų skaičius (vnt.)
NUMCEPS       = 19 # Kepstro koeficientų skaičius
deltawindow   = accwindow = 2 # Skirtuminių požymių skaičiavimo langas
cmvn_lc       = 150 # Kadrų kiekis iš kairės pusės nuo slankiojančio lango esa-
mos padėties
cmvn_rc       = 150 # Kadrų kiekis iš dešinės pusės nuo slankiojančio lango esa-
mos padėties
fs            = 1e7/SOURCERATE # Imties dažnis (Hz)
window        = WINDOWSIZE/SOURCERATE # Kadro lango ilgis atskaitomis
```

```
noverlap      = (WINDOWSIZE-TARGETRATE)/SOURCERATE # Persidengimų tarp gretimų
kadrų ilgis atskaitomis
ESCALE        = 0.1 # logaritminės energijos mastelis
SILFLOOR      = 50.0 # Energijos apatinė slenkstinė riba (dB)

## Loginiai parametrai
ZMEANSOURCE   = True # Vidurkio atėmimas iš signalo
USEHAMMING    = True # Hammingo lango naudojimas
RAWENERGY     = True # Energijos požymio gražinimas
ENORMALISE    = True # Energijos normavimas

## Inicializuojami logaritminių tikėtinumų parametrai
log_likelihood_1_model     = []
log_likelihood_2_model     = []
log_likelihood_gmm_model   = []
log_likelihood_kmean_model = []

## Pagalbinės funkcijos
def load_ubm(fname):
    gmm = np.loadtxt(fname, dtype=np.float32)
    n_superdims = (gmm.shape[1] - 1) // 2
    weights = gmm[:,0]
    means   = gmm[:,1:(n_superdims+1)]
    covs    = gmm[:,(n_superdims+1):]
    return weights, means, covs

def compute_vad(s, win_length=160, win_overlap=80):
    v = evad.compute_vad(s,win_length=win_length, win_overlap=win_overlap, n_re-
alignment=10)
    n_frames = sum(v)
    n_regions = n_frames
    return v, n_regions, n_frames

## Parengiamasis duomenų apdorojimas
# kviečiama Mel filtrų rinkinio apskaičiavimo funkcija
fbank_mx      = MFCC_pozymiai.mel_fbank_mx(winlen_nfft = window,
                                           fs          = fs,
                                           NUMCHANS    = NUMCHANS,
                                           LOFREQ      = LOFREQ,
                                           HIFREQ      = HIFREQ)
# kviečiama UBM failo nuskaitymo funkcija
ubm_file_weights, ubm_file_means, ubm_file_covs = load_ubm(ubm_file)

# Vykdoma GMM objekto inicijavimo funkcija
model_1 = gmm.gmm_eval_prep(ubm_file_weights, ubm_file_means, ubm_file_covs)
model_2 = gmm.gmm_eval_prep(kmean_weights, kmeans_en, kmean_covs)
GMM     = gmm.gmm_eval_prep(ubm_weights, ubm_means.T, ubm_covs.T)
KMEAN   = gmm.gmm_eval_prep(kmean_weights, kmean_means, kmean_covs)

# Vykdomas UBM statistikų normavimas jei UBM vidurkių matricos ubm_menas stul-
pelių skaičius atitinka UBM kovariacijų matricos ubm_covs stulpelių skaičių
dimF=ubm_means.shape[1] # Randamas UBM vidurkių matricos ubm_menas stulpelių
skaičius
if ubm_covs.shape[1] == dimF:
    ubm_norm = 1/np.sqrt(ubm_covs);
if ubm_file_covs.shape[1] == dimF:
    ubm_file_norm = 1/np.sqrt(ubm_file_covs);

for records in FILES:
    ## Vykdomas įrašo nuskaitymas
```

```
    print('records = %s' %records) # Spausdinamas įrašo pavadinimas iš sąrašo
FILES
    rate,sigraw = spiowav.read(records) # Vykdomas įrašo nuskaitymas (dažnis,
signalas)

    ## Vykdoma įrašo duomenų konfigūracija
    # Grąžinamas pirmasis signalo sigraw stulpelis jei signalo dydis ne mažesnis
nei 1
    if len(sigraw.shape)>1:
        sigraw = sigraw[:,0]
    # Pakeičiamas signalo atskaitų dažnis į 8 kHz jei dažnis nelygus 8 kHz
    if rate != 8000:
        secs = len(sigraw)/rate
        samps = secs*8000
        sigraw = scsig.resample(sigraw, samps)
    sig = np.float64(sigraw);

    ## Randami ir spausdinami: dažnis, signalo ilgis, MFCC požymiai (kviečiama
MFCC požymių apskaičiavimo funkcija)
    print('rate ', rate)
    print('sig length ', len(sig))
    print('  Extracting features'),
    fea = MFCC_pozymiai.mfcc_htk(sig,
                window      = np.int32(window),
                noverlap    = np.int32(noverlap),
                fbank_mx    = fbank_mx,
                _0          = 'first',
                NUMCEPS     = NUMCEPS,
                RAWENERGY   = RAWENERGY,
                PREEMCOEF   = PREEMCOEF,
                CEPLIFTER   = CEPLIFTER,
                ZMEANSOURCE = ZMEANSOURCE,
                ENORMALISE  = ENORMALISE,
                ESCALE      = ESCALE,
                SILFLOOR    = SILFLOOR,
                USEHAMMING  = USEHAMMING)

    ## MFCC papildymas pirmos ir antros eilės skirtuminiais požymiais
    fea = MFCC_pozymiai.add_deriv(fea,(deltawindow,accwindow)) # prie MFCC pri-
dedami skirtuminiai požymiai
    ## MFCC išrikiavimas pagal SFEaCut
    print('  Reshaping to SFeaCat convention')
    fea = fea.reshape(fea.shape[0], 3, -1).trans-
pose((0,2,1)).reshape(fea.shape[0],-1) #išrikiuojami MFCC požymiai pagal SFeaCut

    ## Vykdomas aktyvaus kadro signale aptikimas
    print('  Computing VAD ') # spausdinama: skaičiuojamas kadrų aktyvumas
    vad,n_regions,n_frames = compute_vad(sig, win_length=np.int32(window),
win_overlap=np.int32(noverlap)) [:len(fea)]     # Kviečiama aktyvaus kadro apti-
kimo funkcija

    ## Atliekamas požymių normavimas
    fea_03 = MFCC_pozymiai.cmvn_floating(fea[:,[0,3]], cmvn_lc, cmvn_rc, unbia-
sed=True) # atskiriami MFCC 0 ir 3 požymiai ir normuojami pagal failo duomenis -
kviečiama vidurkio ir dispersijos normavimo slankiojančio lango atžvilgiu funk-
cija)
    fea_rest = fea[:,columns_idx] # naujai matricai priskiriami likę požymiai
(be 0 ir 3)
    fea2_rest = (fea_rest - fmean) / smean # likusios požymius normuoja pagal
statistinius duomenis
```

```
    fea2 = np.hstack( (fea_03[:,0][:,None], fea2_rest[:,[0,1]],
fea_03[:,1][:,None], fea2_rest[:,columns_rest_idx]) ) # požymiai sustatomi anks-
tesniąja tvarka

    ## Atliekamas balso sričių požymių sujungimas
    mask_power = fea[:,0]>30 # Sukuriama nauja loginė matrica, kuri teisinga,
kai MFCC matricos pirmo stulpeliai didesni už 30
    mask = (vad==0) * mask_power # randama balso sričių indeksų "kaukė"
    fea2 = fea2[mask,:] # randami balso sričių požymiai

    ## Apskaičiuojami įvairių modelių logaritminiai tikėtinumai
    # 1 modelio
    llh_1 =  gmm.gmm_eval(fea2, model_1, return_accums=0)
    llh_1_mean = np.nanmean(llh_1)
    log_likelihood_1_model = np.append(log_likelihood_1_model, llh_1_mean)
    # 2 modelio
    llh_2 =  gmm.gmm_eval(fea2, model_2, return_accums=0)
    llh_2_mean = np.nanmean(llh_2)
    log_likelihood_2_model = np.append(log_likelihood_2_model,llh_2_mean)
    # pasirinktos kalbos GMM modelio
    llh_gmm =  gmm.gmm_eval(fea2, GMM, return_accums=0)
    llh_gmm_mean = np.nanmean(llh_gmm)
    log_likelihood_gmm_model = np.append(log_likelihood_gmm_model, llh_gmm_mean)
    # pasirinktos kalbos k-vidurkių modelio
    data_sqr = fea2[:, KMEAN['utr']] * fea2[:, KMEAN['utc']]
    gamma = -0.5 * data_sqr.dot(KMEAN['invCovs'].T) + fea2.dot(KMEAN['in-
vCovMeans'].T) + KMEAN['gconsts']
    xmax = gamma.max(0)
    ex=sp.exp(gamma - np.expand_dims(xmax, 0))
    llh_kmean = xmax + sp.log(sp.sum(ex, 0))
    not_finite = ~np.isfinite(xmax)
    llh_kmean[not_finite] = xmax[not_finite]
    llh_kmean_mean = np.nanmean(llh_kmean)
    log_likelihood_kmean_model = np.append(log_likelihood_kmean_mo-
del,llh_kmean_mean)

# Rezultatų išsaugojimas
np.save('german\\log_likelihood_1_model', log_likelihood_1_model) # (pakeisti
pagal kalbą)
np.save('german\\log_likelihood_2_model', log_likelihood_2_model) # (pakeisti
pagal kalbą)
np.save('german\\log_likelihood_9_model', log_likelihood_gmm_model) # (pakeisti
pagal kalbą)
np.save('german\\log_likelihood_10_model', log_likelihood_kmean_model) # (pa-
keisti pagal kalbą)
```

STATISNINIŲ TESTŲ PUSPROGRAMĖ

```
# Įvesties duomenys
en1 = np.load('english\\log_likelihood_1_model.npy')
en2 = np.load('english\\log_likelihood_2_model.npy')

esp1 = np.load('spanish\\log_likelihood_1_model.npy')
esp2 = np.load('spanish\\log_likelihood_2_model.npy')
esp3 = np.load('spanish\\log_likelihood_3_model.npy')
esp4 = np.load('spanish\\log_likelihood_4_model.npy')

it1 = np.load('italian\\log_likelihood_1_model.npy')
it2 = np.load('italian\\log_likelihood_2_model.npy')
it5 = np.load('italian\\log_likelihood_5_model.npy')
it6 = np.load('italian\\log_likelihood_6_model.npy')

fr1 = np.load('french\\log_likelihood_1_model.npy')
fr2 = np.load('french\\log_likelihood_2_model.npy')
fr7 = np.load('french\\log_likelihood_7_model.npy')
fr8 = np.load('french\\log_likelihood_8_model.npy')

ru1 = np.load('russian\\log_likelihood_1_model.npy')
ru2 = np.load('russian\\log_likelihood_2_model.npy')
ru9 = np.load('russian\\log_likelihood_9_model.npy')
ru10 = np.load('russian\\log_likelihood_10_model.npy')

de1 = np.load('german\\log_likelihood_1_model.npy')
de2 = np.load('german\\log_likelihood_2_model.npy')
de11 = np.load('german\\log_likelihood_11_model.npy')
de12 = np.load('german\\log_likelihood_12_model.npy')

lyginbamos_kalbos = (('anglų/ispanų', 'anglų/italų', 'anglų/prancūzų',
'anglų/rusų', 'anglų/vokiečių'))

en100_1 = np.load('final testing records\\english_log_likelihood_1_model.npy')
en100_2 = np.load('final testing records\\english_log_likelihood_2_model.npy')
esp100_1 = np.load('final testing records\\spanish_log_likelihood_1_model.npy')
esp100_2 = np.load('final testing records\\spanish_log_likelihood_2_model.npy')

p1=np.zeros((5,))
p2=np.zeros((5,))
stat1=np.zeros((5,))
stat2=np.zeros((5,))

## Vieno veiksnio dispersinė analizė (veiksnys kalba)
F1, p_1 = stats.f_oneway(en1, esp1, it1, fr1, ru1, de1)
F2, p_2 = stats.f_oneway(en2, esp2, it2, fr2, ru2, de2)
F_gmm, p_gmm = stats.f_oneway(esp3, it5, fr7, ru9, de11)
F_kmean, p_kmean = stats.f_oneway(esp4, it6, fr8, ru10, de12)

## t-testas
# 1 modelio
stat1[0], p1[0] = stats.ttest_ind(en1, esp1, equal_var = False) # anglų/ ispanų
stat1[1], p1[1] = stats.ttest_ind(en1, it1, equal_var = False) # anglų/ italų
stat1[2], p1[2] = stats.ttest_ind(en1, fr1, equal_var = False) # anglų/ prancūzų
stat1[3], p1[3] = stats.ttest_ind(en1, ru1, equal_var = False) # anglų/ rusų
stat1[4], p1[4] = stats.ttest_ind(en1, de1, equal_var = False) # anglų/ vokiečių
```

```
# 2 modelio
stat2[0], p2[0] = stats.ttest_ind(en2, esp2, equal_var = False) # anglų/ ispanų
stat2[1], p2[1] = stats.ttest_ind(en2, it2, equal_var = False) # anglų/ italų
stat2[2], p2[2] = stats.ttest_ind(en2, fr2, equal_var = False) # anglų/ prancūzų
stat2[3], p2[3] = stats.ttest_ind(en2, ru2, equal_var = False) # anglų/ rusų
stat2[3], p2[3] = stats.ttest_ind(en2, de2, equal_var = False) # anglų/ vokiečių

p_model_1 = np.transpose(np.vstack(([lyginbamos_kalbos], [p1])))
p_model_2 = np.transpose(np.vstack(([lyginbamos_kalbos], [p2])))

## t-testas, kai lyginamos kalbos - anglų/ispanų. Testuojama po 100 įrašų
stat100_1, p100_1 = stats.ttest_ind(en100_1, esp100_1, equal_var = False) # 1
modelio
stat100_2, p100_2 = stats.ttest_ind(en100_2, esp100_2, equal_var = False) # 2
modelio

# Rezultatų išsaugojimas
np.save('Statistiniu testų rezultatai\\p_model_1', p_1)
np.save('Statistiniu testų rezultatai\\p_model_2', p_2)
np.save('Statistiniu testų rezultatai\\p_gmm_models', p_gmm)
np.save('Statistiniu testų rezultatai\\p_kmean_models', p_kmean)
np.save('Statistiniu testų rezultatai\\p_en_fr_100_1', p100_1)
np.save('Statistiniu testų rezultatai\\p_en_fr_100_2', p100_2)
np.save('Statistiniu testų rezultatai\\p_t-test_model_1', p_model_1)
np.save('Statistiniu testų rezultatai\\p_t-test_model_2', p_model_2)
```