KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS
INFORMATIKOS STUDIJŲ PROGRAMA

ROBERTAS BIVAINIS

BALSO ATPAŽINIMO PROGRAMŲ LIETUVINIMO GALIMYBIŲ TYRIMAS

Magistro baigiamasis darbas

Vadovas
doc. dr. S. Drąsutis

Konsultantas
doc. dr. V. Rudžionis

KAUNAS, 2013

KAUNO TECHNOLOGIJOS UNIVERSITETAS
INFORMATIKOS FAKULTETAS
INFORMATIKOS STUDIJŲ PROGRAMA

ROBERTAS BIVAINIS

BALSO ATPAŽINIMO PROGRAMŲ LIETUVINIMO GALIMYBIŲ TYRIMAS

Magistro baigiamasis darbas

Darbo vadovas
doc. dr. S. Drąsutis

Konsultantas
doc. dr. V. Rudžionis

Recenzentas
doc. dr. K. Ratkevičius

KAUNAS, 2013

AUTORIŲ GARANTINIS RAŠTAS
DĖL PATEIKIAMO KŪRINIO

2013 - 06 - 03 d.
Kaunas

Autoriai,_____
(vardas, pavardė)

_____,

patvirtina, kad Kauno technologijos universitetui pateiktas baigiamasis bakalauro (magistro) darbas (toliau vadinama – Kūrinys)_____
(kūrinio pavadinimas)
_____

pagal Lietuvos Respublikos autorių ir gretutinių teisių įstatymą yra originalus ir užtikrina, kad

1) jį sukūrė ir parašė Kūrinyje įvardyti autoriai;
2) Kūrinys nėra ir nebus įteiktas kitoms institucijoms (universitetams) (tiek lietuvių, tiek užsienio kalba);
3) Kūrinyje nėra teiginių, neatitinkančių tikrovės, ar medžiagos, kuri galėtų pažeisti kito fizinio ar juridinio asmens intelektinės nuosavybės teises, leidėjų bei finansuotojų reikalavimus ir sąlygas;
4) visi Kūrinyje naudojami šaltiniai yra cituojami (su nuoroda į pirminį šaltinį ir autorių);
5) neprieštarauja dėl Kūrinio platinimo visomis oficialiomis sklaidos priemonėmis.
6) atlygins Kauno technologijos universitetui ir tretiesiems asmenims žalą ir nuostolius, atsiradusius dėl pažeidimų, susijusių su aukščiau išvardintų Autorių garantijų nesilaikymu;
7) Autoriai už šiame rašte pateiktos informacijos teisingumą atsako Lietuvos Respublikos įstatymų nustatyta tvarka.

Autoriai

_____ (vardas, pavardė)	_____ (parašas)
_____ (vardas, pavardė)	_____ (parašas)
_____ (vardas, pavardė)	_____ (parašas)
_____ (vardas, pavardė)	_____ (parašas)

SANTRAUKA

Bivainis, R. Balso atpažinimo programų lietuvinimo galimybių tyrimas: Informatikos magistro baigiamasis darbas / vadovas doc. dr. S. Drąsutis; Kauno technologijos universitetas, Informatikos fakultetas, Multimedijos inžinerijos katedra. Kaunas, 2013. 50 p.

Šiuolaikiniame pasaulyje atsiranda vis daugiau balso atpažinimo programų anglų ir kitomis kalbomis. Sprendžiant kalbos atpažinimo klausimą buvo įdėta nemažai pastangų kuriant ir tobulinant anglų ir kitų kalbų atpažinimo sistemas, tačiau lietuvių kalbos atpažinimas yra dar neįminta mįslė ir visi esami ir kuriami produktai yra dar tik bandymų ir lygmenyje. Taigi, šiame darbe bus bandoma patobulinti (tiksliau siekiama padėti komandai dirbančiai su lietuvių kalbos balso atpažinimo sistemomis sukurti realų produktą). Visa informacija ir algoritmai, kuriais remiamasi kuriant lietuvių kalbos balso atpažinimo programą bus renkami iš jau laiko patikrintų ir esamų anglų kalbos mokslinių darbų straipsnių, knygų. Tačiau būtina paminėti, kad nepaisant akivaizdžios anglų kalbos atpažinimo programų pažangos net ir geriausių automatinių atpažinimo sistemų efektyvumas yra prastesnis negu žmogaus (klausytojo) atpažinimo tikslumas, o nepalankioje aplinkoje - dar blogesnis. Taigi dar yra tikrai daug erdvės tokių sistemų tobulinimui, o ypač kuriant sistemas skirtas atpažinti lietuvių kalbai.

Šiame darbe yra analizuojama ir tiriama kaip veikia balso atpažinimo sistema HTK, kokie žingsniai turi būti atlikti norint sėkmingai atpažinti lietuviškai išartus žodžius. Taip pat apžvelgiamos kokių kalbos technologijų samprata reikalinga norint sukurti balso atpažinimo programą.

Balso atpažinime labai svarbu yra kalbos signalų atpažinimo modeliai ir paslėptosios Markovo grandinės, todėl analizėje yra apžvelgiama jų veikimo principai ir algoritmai.

Atliekant eksperimentą buvo naudojama HTK balso atpažinimo sistema. Buvo pasirinkta 25 medicininiai terminai, kurie buvo sunkiai atpažįstami kitų programų. Programai buvo paruoštas gramatikos failas, skaičiuojami požymių failai taip pat parengiami modelių failai.

Norint, kad sistema veiktų sėkmingai, 12 diktorių buvo įrašyti 6000 medicininių terminų, kad sistema turėtų tam tikrą duomenų ir skirtingų diktorių bazę. Pagal įrašytas frazes buvo parengti modelių failai, apmokyti ir atliktas testavimas įrašius tas pačias frazes savo balsu.

Išanalizavus HTK balso atpažinimo programos veikimo principą, paaiškėjo, jog norint, kad sistema atpažintų tam tikrą žodį, reikia parengti gramatikos failą, kuriame turime nurodyti žodžių bazę, taip pat parengti script failus, kuriuose reikia nurodyti kelius, kur patalpinti įrašai, taip pat parengti modelių failus.

Kadangi HTK sistemoje naudojami ne patys įrašai, o iš tų įrašų apskaičiuoti požymiai buvo apskaičiuoti požymiai ir padaryti script-failai, kad sistema sugebėtų rasti kur yra įrašų failai. Tokiu būdu HTK sistema lengvai suranda esamus įrašus, apdoroja juos ir atpažįsta.

Įrašius žodžius kurie buvo apmokyti pagal kitus diktorius. Bendra visų atpažintų žodžių procentas yra 18,4%.

Atlikti bandymai ir pateikiami rezultatai su ispanų kalbos atpažintuvu *Microsoft Speech Recognizer 8.0*

SUMMARY

Bivainis, R. Speech Recognition Program's Lithuanization possibility survey: Informatics Master Thesis/ Supervisor doc. Dr. S. Drąsutis: Kaunas University of Technology, Faculty of Informatics; Cathedral of Multimedia Engineering. Kaunas, 2013. 45 p.

Nowadays more and more speech recognition programs see the world. A lot of effort was put in creating these programs in various languages, however speech recognition systems in Lithuanian language is not a sufficiently explored field. All such created products are still on the level of development ant trial. This thesis will help to improve the development of speech recognition programs in Lithuanian language. All information and algorithms used in this thesis are based on various research studies and books. It is worth mentioning that even most advanced speech recognition programs in English language are still faulty and an unfavorable environment makes it even harder for them to recognize the speech. There is a lot of room for development, especially in creating speech recognition programs in Lithuanian language.

This thesis will focus on how the speech recognition program HTK operates and what steps have to be taken in order to recognize spoken Lithuanian words. Also the emphasis of this thesis goes to conceptions of speech recognition technologies which are needed to create a speech recognition program.

The first part of this thesis overviews relation between phonology and phonetics and the conception of sound. Speech recognition is also related to human's physiological structure like vocal tract structure and human speech mechanism, therefore these subjects are also reviewed in the thesis along with noise and echo reduction which is also very important in recognizing the speech.

Two very important things in voice recognition are the speech recognition models and the hidden Markov chains. This thesis includes the overview of the aforementioned models, their operating principles and algorithms.

During the experiment the speech recognition program HTK was used. 25 medical terms were selected. These terms were hardly recognized by other speech recognition programs. A special grammar file, counting sign files and special model files were prepared for this experiment.

6000 medical terms were recorded in 12 speakers in order to create a data base and a different speaker base. Certain model files were created using recorded phrases. The test was conducted using recorded medical terms vocally.

TURINYS

LENTELIŲ SĄRAŠAS

PAVEIKSLŲ SĄRAŠAS

ĮVADAS

Šiuolaikiniame pasaulyje atsiranda vis daugiau balso atpažinimo programų anglų ir kitomis kalbomis. Sprendžiant kalbos atpažinimo klausimą buvo įdėta nemažai pastangų kuriant ir tobulinant anglų ir kitų kalbų atpažinimo sistemas, tačiau lietuvių kalbos atpažinimas yra dar neįminta mįslė ir visi esami ir kuriami produktai yra dar tik bandymų lygmenyje. Taigi, šiame darbe bus bandoma patobulinti (tiksliau siekiama padėti komandai dirbančiai su lietuvių kalbos balso atpažinimo sistemomis sukurti realų produktą). Visa informacija ir algoritmai, kuriais remiamasi kuriant lietuvių kalbos balso atpažinimo programą bus renkami iš jau laiko patikrintų ir esamų anglų kalbos mokslinių darbų straipsnių, knygų. Tačiau būtina paminėti, kad nepaisant akivaizdžios anglų kalbos atpažinimo programų pažangos net ir geriausių automatinių atpažinimo sistemų efektyvumas yra prastesnis negu žmogaus (klausytojo) atpažinimo tikslumas, o nepalankioje aplinkoje - dar blogesnis. Taigi dar yra tikrai daug erdvės tokių sistemų tobulinimui, o ypač kuriant sistemas skirtas atpažinti lietuvių kalbai.

- Taigi pradžioje būtina aptarti visas svarbiausias balso technologijų grupes. Visų pirmiausia išskirsime tris pagrindines balso technologijų grupes:
- balsu tariamų vienetų (žodžiai, jų sekos, frazės) automatinis nustatymas arba kalbos atpažinimas;
- teksto skaitymas balsu (teksto sintezė);
- kitos balso technologijos (asmens tapatybės vertinimas pagal jo balsą, kalbos signalų suspaudimas, triukšmų slopinimas ir pan.).

Atpažinimas, kurio pagrindinė paskirtis yra automatiškai nustatyti, kas yra sakoma informacijos priėmimo sistemai. Tai gali būti atskiras žodis (balso komanda), žodžių seka (PIN kodas), net rišlių sakinių skaitymas. Informacijos priėmimo sistema, nustačiusi, kas jai buvo pasakyta, atlieka atitinkamus veiksmus.

Sintezė, kurios pagalba pagal reikiamą komandą balsu perskaitoma informacinėje sistemoje teksto pavidalu saugoma informacija. Sintezei priskiriamos ir paprastesnės informacijos pateikimo balsu formos, pvz., iš anksto paruoštų žodžių ar jų sekų pateikimas balsu, esant tam tikram reikalavimui.

Kitos balso technologijos, tokios, kaip asmens tapatybės vertinimas pagal jo balsą reikalingas teisėsaugoje ir komercinių operacijų vykdymui. Triukšmų šalinimas nuo kalbos signalų yra priemonė sukauptoms kultūros vertybėms restauruoti. Iš esmės triukšmų apdorojimo problemos liečia beveik visas kitas balso technologijas. Balso signalų suspaudimas yra taupaus balso įrašų saugojimo ar perdavimo priemonė operuojant balsu internete.

Temos aktualumas – naujumas. Praktikoje dar labai mažai programų, kurios naudojamos atpažinti lietuviškus tekstus, todėl šiuo darbu stengtasi bent kiek užpildyti panašaus pobūdžio tyrimų ir jų įgyvendinimo spragas. Įsigilinti kodėl reikia lietuviškų balso atpažinimo programų ir kaip jas pritaikyti praktikoje. Darbe bandoma analizuoti ir išsiaiškinti, kaip kitą kalbą atpažįstančią programą pakeisti tokia pačia lietuviška. Turbūt aktualiausia šios balso atpažinimo programos pritaikymo sritis pramonėje t.y. realiame gyvenime yra pagalba žmonėms, kurie serga sunkiomis įgimtomis ligomis ar yra paralyžiuoti cerebriniu paralyžiumi ir negalim valdyti savo kūno dalių. Tokia balso atpažinimo programą turėtų pagelbėti norint naudotis kompiuteriu ir ne tik, reikėtų viską valdyti balsu.

Tyrimo (darbo) objektas. Balso atpažinimo programa.

Darbo tikslas. – atlikti vaistų balso komandų atpažinimo kokybės tyrimą, sudarius daugiadiktorinį garsyną, paruošiant naujus gramatikų rinkinius su atitinkamomis transkripcijomis bei paruošiant demonstracinę programą realiam pritaikymui.

Siekiant darbo tikslo, numatyti **uždaviniai:**

Teoriškai pagrįsti, atlikti literatūros šaltinių analizę, šnekamosios kalbos atpažinimo, kalbos sintezavimo, garsynų, gramatikų sudarymo ir transkripcijų temomis sudaryti daugiadiktorinį vaistų garsyną bei suprojektuoti ir parengti modelių failus atpažinimo sistemai

Paruošti gramatikos failus HTK atpažinimo sistemai, kurie bus naudojami tyrime;

Paruošti HTK atpažintuvą balso komandų atpažinimo tyrimui bei ištirti jam naudojamų profilių darbą skirtinguose darbo režimuose bei veikimo principus;

Atlikti kokybinę rezultatų analizę ir palyginti pradinius bei galutinius rezultatus;

Įsisavinti ispaniško sintezatoriaus atpažinimo programų paketo veikimą ir atlikti dalies garsyno tyrimą;

Paruošti demonstracinę programą, kuri sietųsi su naršykle ir atskleistų realias vaistų balso komandų pritaikymo galimybes.

Įrašyti žodžius savo balsu ir juos apmokyti ir atlikti testavimą su pasirinktais modeliais;

Aprašyti, kaip skirtingi metodai ir kitos aplinkybės gali lemti atpažinimo tikslumą, pagal gautus rezultatus.

Darbo struktūra: Darbą sudaro įvadas, teorinė ir analizės dalis, metodologinė dalis, tiriamoji – eksperimentinė dalis, išvados, literatūros sąrašas, 4 priedai (vienas jų t.y. HTK paketo įrankiai pateikiami elektroninė versija). Teorinę analizės dalį sudaro 20 puslapių, praktinę eksperimentinę dalį – 14 puslapių. Literatūros sąrašą sudaro 22 šaltiniai.

1. KALBOS TECHNOLOGIJOS

Šiame darbe bus analizuojama ir pateikiama realių pavyzdžių kaip veikia įvairios balso atpažinimo programos ir jų pagrindu sukurti arba pritaikyti esamą balso atpažinimo programą lietuvių kalbai. Kadangi balso atpažinimo programa neįsivaizduojama be kalbos, pradžioje pateiksiu informacijos apie tai, kas yra ta kalba ir balsas.

Verbalinės komunikacijos modeliai ir kalbos funkcijos struktūriškai daug kuo panašūs į bendruosius komunikacijos modelius, kad ir kokie jie būtų – techniniai elektroniniai mechanizmai ar gyvųjų organizmų informacijos cirkuliavimo sistemos. Bet verbalinės komunikacijos modeliai pasižymi vienu labai svarbiu bruožu – jie yra funkciniai [[4]]. Aristotelis savo „Retorikoje: aprašė, kad kalba pagrįsta pagal žmonių bendravimo pamatus. Pagal Aristotelį retorinę situaciją sudaro trys elementai: oratorius, kalba ir klausytojas. Bet retorika esanti tam, toliau teigia Aristotelis, kad veiktų nutarimų priėmimą, kitaip tariant, kad įtikintų klausytoją. Toks įsitikinimas yra trejopas. Vieną lemia oratoriaus charakteris, kitas susijęs su psichine klausytojo būkle, trečiąjį sudaro tikrasis arba tariamas kalbos įrodymas. Išskyręs tris įrodinėjimo aspektus, Aristotelis kartu nustatė taip pat ir funkcijas, kurias atlieka kiekviena sudėtinė retorinio akto struktūros dalis [2]. Vieną iš pirmųjų verbalinės komunikacijos modelių sukūrė vokiečių mokslininkas [2]. Juo siekta apibrėžti kalbos funkcijas. Jo sukurtas kalbos modelis yra vadinamas instrumentiniu, arba kalbos organo modeliu. Tai, pirma, šnekos (kalbėjimo) įvykio, antra, kalbos funkcionavimo ir, trečia, kalbos ženklo modelis. Pagal šį modelį, kiekvienas šnekos aktas, arba šnekos įvykis, kiekvienas verbalinės komunikacijos aktas susideda iš tokių pastovių dalių: siuntėjas, arba adresantas, siunčia pranešimą gavėjui, arba adresatui. Kad pranešimas būtų veiksmingas, reikia nurodomojo konteksto (kitais, dviprasmiškais žodžiais tariant, referento), kuris būtų verbalinis arba verbalizuojamas ir adresatui suprantamas. Pasak Jakobsono (Jakobson), verbalinės komunikacijos akto struktūra bendrais bruožais yra tokia: 1) *pranešimas* perduodamas 2) *šnekos signalų* – akustinių ar vaizdinių – virtinėmis. Nešančių informaciją signalų virtinę 3) adresantas, kalbantysis ar rašantysis, siunčia 4) adresatui, klausančiajam arba skaitančiajam, tam tikru 5) ryšio kanalu (oru, kuriuo sklinda garsas, telefono laidu, popieriumi ir kt.). Iš adresanto pasiųstų signalų adresatas išgauna tą pačią ar maždaug tą pačią informaciją, kurią adresantas turėjo galvoje, dėl to, kad ir vienas, ir antras turi bendrą ar iš dalies bendrą 6) kodą, t.y. (šnekos) signalų ir informacijos atitikimų taisykles.

Daug užsienio ir Lietuvos mokslininkų šiandien nagrinėja su kalbos technologijomis susijusias sritis. Šių tyrimų svarba yra nenusakomai didelė. Kaip teigia vienas iš pagrindinių Microsoft kompanijos vykdančiųjų veikėjų S. Ballmer. Informacinių technologijų nauda visuomenei yra akivaizdi: naujai kuriamos, atnaujinančios technologijos įgalina žmones išlaisvinti savo potencialą, suteikia galimybę daryti tai, ko jie iki šiol nė nenumanė galintys daryti. Štai kodėl yra labai svarbūs tyrinėjimai šiose srityse. Ne paslaptis, kad šie tyrimai yra svarbūs ir dėl to, kad auga vartotojų poreikiai, naujausios technologijos užkariauja mūsų kasdienybę. Kaip yra pasakęs Microsoft kompanijos kūrėjas B. Gates „Popierius jau senai nėra didžioji mano dienos darbų dalis“ [17]. Tokie ir panašūs teiginiai iliustruoja informacinių technologijų, tai pat ir kalbos technologijų, plėtros svarbą.

Pasak R. Maskeliūno [2]. Šnekamosios kalbos efektyvumas yra stulbinantis, neginčijamai kalbai tai pats natūraliausias bendravimo būdas. Tad kaipgi technologijos, be šnekamosios kalbos integravimo. Kaip teigia K. Driunys kalbos technologijos sparčiai integruojasi ir tobulina informacinių technologijų sritį. Vis dažniau šios technologijos panaudojamos praktiškai, įvairioms programoms kurti. Žinoma, akivaizdu yra ir tai, kad dar kol kas nei viena iš šnekamosios kalbos atpažinimo algoritmų sistemų neprilygsta natūraliems žmonių sugebėjimams apdirbti, suvokti ir generuoti kalbą. Tad labai svarbūs yra tyrimai, galintys padėti tobulinti ir atnaujinti šias technologijas, tam, kad būtų galima tikslingai žengti tobuliausios sistemos kūrimo link. Kalbinis dialogas su kompiuteriu. Tai yra tiesioginis bendravimas su kompiuteriu balso pagalba. Tam pritaikytose programose yra paruoštas komandų sąrašas ir, vartotojui ištarus vieną iš jų, kompiuteris,

priklausomai nuo atpažintos komandos numerio, atitinkamai reaguoja. Komandos gali būti labai įvairios pvz.: “parodyk kiek valandų”, “pasakyk kiek valandų”, “pagrok Lietuvos himną” ir t.t. programai nepavykus patikimai atpažinti balso komandos, išvedamas pranešimas apie tai ir prašymas pakartoti komandą. Atpažinus baigimo komandą, programa baigia darbą [5].

Balso įrašų stenografavimas. Programinė įranga, skirta balso įrašams stenografuoti naudojama interviu, posėdžių, forumų, sesijų, derybų, paskaitų, mitingų, konferencijų metu. Šiose programose kartu su tekstiniu redaktoriumi paleidžiama foninė garsinio išvedimo programa, valdoma iš teksto redaktoriaus aplinkos su makrofunkcijomis. Vartotojas gali sustabdyti išvedimą bet kuriuo laiko momentu bei tęsti nuo sustabdymo vietos, valdyti išvedimo greitį, pakartoti tam tikro įrašo fragmento išvedimą, automatiškai daryti pageidaujamo dydžio pauzes kas apibrėžtą laiko intervalą, keisti išvedimo programos parametrus nepertraukiant darbo.

Balsinė interneto naršyklė. Interneto naršymas balsu yra pakankamai patogus - tam skirtos programos gali atlikti platu spektrą komandų: paleisti naršyklė, padaryti naršyklė matoma, užkrauti nurodytą interneto svetainę, uždaryti naršyklę, gauti tekstą, kuris yra užkrautoje interneto svetainėje, gauti užkrautos svetainės HTML kodą, stabdyti svetainės atidarymą, pakartotinai užkrauti svetainę, atidaryti namų svetainę, atidaryti sekančią svetainę, atidaryti prieš tai buvusią svetainę. Interneto svetainių skaitymo programos pagrindinis langas yra mažo formato, kad netrukdytų vartotojui stebėti interneto svetainių.

Balsas namų automatikoje. Labai patogu ir praktiška buitinius elektros prietaisus įjungti ir išjungti balsu. Programinę įrangą sudaro personaliniame kompiuteryje veikianti komandų atpažinimo programa bei valdiklio programa, per nuoseklų prievadą iš kompiuterio priimanti atpažintos komandos numerį ir per I2C magistralę atliekanti elektrinių įrenginių įjungimą/išjungimą.

Asmens atpažinimas pagal jo balsą. Kaip jau minėta, tai labai svarbu kriminalistikoje. Moksliniuose kalbinių technologijų tyrinėjimuose kalbėtojo atpažinimas pagal jo balsą yra suprantamas kaip procesas, kurio metu iš asmens kalbos signalų išskiriami identifikaciniai požymiai, pagal kuriuos ir atpažįstamas konkretus asmuo. Kalbėtojo atpažinimas savo ruožtu skirstomas į identifikavimą ir verifikavimą [13]. Tiek verifikavimas, tiek identifikavimas gali būti priklausomas arba nepriklausomas nuo teksto. Teismo ekspertizėje asmens atpažinimas pagal balsą vadinamas asmens identifikavimu ir dažniausiai yra nepriklausomas nuo teksto. Kriminalistikoje naudojami ir verifikavimo elementai, kai atliekamas asmens vertinimas pagal balsą. Kaip pavyzdį čia galima paminėti asmens balso palyginimą su balsais iš turimos balsų bazės. Pagrindinis bruožas skiriantis teisminį asmens identifikavimą nuo kitų verifikavimo sistemų yra tai, kad mes turime „nelinkusio bendrauti“ asmens balsą. T.y. dažniausiai tiriamasis ir lyginamasis garso įrašai stipriai skiriasi. Šis skirtumas susidaro dėl skirtingų garso įrašų darymo sąlygų, asmens skirtingų emocinių būsenų, triukšmo įtakos, garso įrašymo kanalų nesutapimo ir t.t. Šiuo metu asmens identifikavime pagal balsą yra naudojami šie pagrindiniai metodai: a) sonografinis metodas, b) fonetinis – akustinis metodas, c) kombinuotas metodas, d) automatinis metodas.

Iš esmės balso atpažinimo technologija yra ne naujas dalykas. Pagrindus šiai technologijai aštuntojo dešimtmečio pradžioje padėjo IBM korporacijos ir Carnegie Mellon universiteto mokslininkai. Nuo to laiko šią sritį ėmė plėtoti įvairių kompanijų ir universitetų tyrėjų grupės. Balso apdorojimo technologijos – tai tokia veiklos sritis, kuri apima įvairiausius mokslus: kompiuteriją, taikomąją matematiką, elektrotechniką, lingvistiką ir informatiką. Kai žmogus šneka, mikrofonas garso bangas verčia analoginiu signalu, o šis keičiamas skaitmeniniu. Iš skaitmeninio signalo kas 10 ar 20 milisekundžių išrenkami informacijos langai. Kiekvienas langas turi būti toks trumpas, kad per jį nekistų informacijos dažninės savybės, ir toks ilgas, kad apimtų bent vieną dažnio periodą. Balso apdorojimo sistema iš kiekvieno lango išrenka tik jai reikalingą spektrinės priklausomybės informaciją, o visa kita – atmeta [14]. Toliau ASR gautus rezultatus palygina su bibliotekoje saugomu žodžiu atitikmenimis. Teoriškai kalbos duomenis būtų galima lyginti su visų bibliotekos žodžių akustine duomenų baze, kurioje atsispindi netgi žodžių tarimo ypatybės - akcentas, tarmė ir t.t. Bet galop, kai bus rasti geriausi ištartų žodžių atitikmenys, pasirodys, kad toks žodžių atpažinimo būdas per lėtas, taigi šiuo būdu užduočių neįmanoma atlikti realiu laiku. Yra ir kitas atpažinimo principas.

Pirmiausia ASR ištartą žodį lygina su kalbos fonemų ir alofonų (poziciniu fonemos variantu) akustine duomenų baze. Fonema yra mažiausias kalbos garsins sistemos vienetas, skiriantis tos kalbos žodžius. Lietuvių kalboje yra 65 fonemos ir keli tūkstančiai alofonų. Taigi, balso atpažinimo sistemos bibliotekos žodžiai gali būti pateikiami pagal fonemų ir alofonų visumą, t. y., pagal žodžiu tarimą. Balso atpažinimo principas gali remtis ir kalbos modeliais, kurie žodžius jungia į frazes ar sakinius. Paprastesniame gramatikos - kalbos modelyje, sistema žodžius atpažįsta tik iš konteksto. Gramatinis modelis gerai veikia dialogo, įsakinėjimo ar valdymo atvejais. Jei nenorima, kad kalbėtojas būtų varžomas būtinybės kalbėti įsakmiai, kalbos modelis gali remtis statistiniais kalbos ypatumais - juk kai kurių žodžių junginiai yra dažnai vartojami. Balso sistemų veiksmingumas priklauso nuo to, kaip sudarytas atpažinimo algoritmas, ir nuo to, kaip tarpusavyje susiję atpažinimo metodai. Pradėti analizuoti balsą galima įvairiais būdais. Tarkim, sistema atpažino kelias fonemas. Toliau balsas gali būti analizuojamas tik tinkamiausiais artiniais. Kartais sistema identifikuoja žodį dar nepasibaigus tam skirtam laikui, nes nustatoma aukšta šio žodžio atpažinimo tikimybė. Kartais analizės pabaigoje sistema nusprendžia, kad identifikavimo tikimybė nėra didelė, arba randa kelis ištartų žodžių atitikmenis [7]. Tokiais atvejais sistema gali paprašyti pakartoti sakini. Triukšmu įtakos mažinimas. Balso atpažinimo technologijas naudinga įdiegti automobiliuose. Tokią įrangą turintį automobilį vairuoti daug saugiau, nes vairuotojas mobiliajam telefonui gali diktuoti, į jį net nepažiūrėjęs, o be to, patogiau valdyti prietaiso skydelį bei orientuotis kelyje. Deja, automobilyje yra daugybę visokių triukšmo šaltinių: variklio ir vėjo gaudesys, išorės bei radijo imtuvo triukšmas, keleivių kalbos. Triukšmo ir balso dažnius nesunku atskirti. Svarbiausia, nustatyti pašalinių garsų (tarp jų ir atsitiktinių žmogaus ištartų žodžių) ypatumus, pagal kuriuos būtų galima atskirti triukšmą nuo balso garsų ir jį filtruoti. Kai kuriuos triukšmus, pvz. įvairiu greičiu važiuojančio automobilio variklio ūžesį, galima išmatuoti iš anksto, kad vėliau būtų nesunku jų atsikratyti.

1.1. Žmonių bendravimas balsu

Pasak Karaliūno šnekamoji kalba yra pagrindinė žmonių komunikavimo forma. Tokio komunikavimo procese galime išskirti kalbos generavimo ir kalbos suvokimo stadijas. Kalbos generavimo procesas prasideda kai kalbėtojas savo mintyse suformuluoja pranešimą, kurį jis nori perduoti klausytojui per kalbą. Sekantis žingsnis yra pranešimo pakeitimas į žodžių seką.

Kiekvienas žodis sudarytas iš fonemų, nusakančių žodžio tarimą. Sakinio prozodija apibrėžia fonemų trukmes, garsumą bei sakinio intonaciją. Kai visa reikalinga informacija (kalbos kodas) paruošiama, kalbėtojas turi atlikti eilę neuronais valdomų raumenų veiksmų, kad galėtum priversti kada reikia vibruoti balso stygas ir suformuoti balso trakto formą taip, kad būtų sukurta ir pasakyta reikalinga kalbos garsų seka ir sukurtas akustinis signalas [5]. Neuronais valdomos raumenų komandos vienu metu valdo visų artikuliatorių judėjimą, taip pat lūpų, žandikaulio, liežuvio ir gomurio. Kalbėtojas kontroliuoja kalbos padargus pagal gaunamą grįžtamą informaciją per savo klausos aparatą. Kai kalbos signalas yra sugeneruotas ir perduotas klausytojui, prasideda kalbos suvokimo procesas. Pirmiausia akustinis signalas yra apdorojamas išilgai vidinės ausies baziliarinės membranos, kuri atlieka ateinančio signalo spektro kitimo laike analizę. Neuroninio pakeitimo procese spektro signalas baziliarinės membranos išėjime pakeičiamas klausos nervo aktyvumo signalais, kas grubiai atitinka požymių išskyrimo procesą. Nervų aktyvumas išilgai klausos nervo smegenyse yra paverčiamas kalbos kodu ir galiausiai yra suvokiamas pranešimas [5].

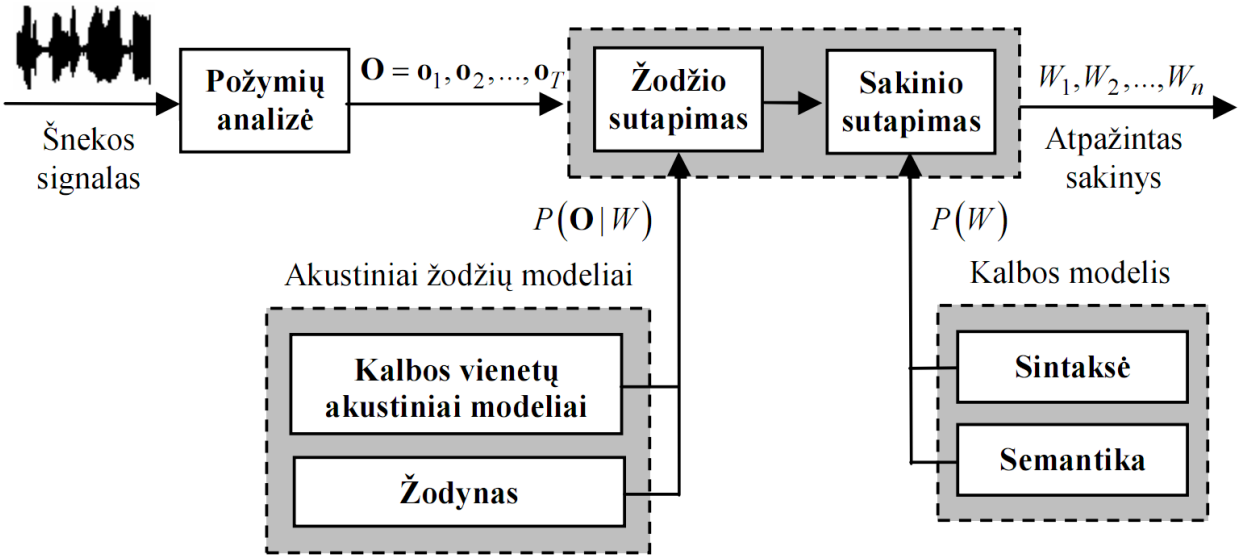
1.2. Kalbos atpažinimas

Manoma, jog įtaką šnekos atpažinimui padarė 1877 metais Tomo Edisono išrastas fonografas – įrenginys, skirtas garso įrašymui ir atkūrimui, dar vadinamas pirmąja pasaulyje „kalbančia mašina“. Šiame XXI informacinių technologijų amžiuje įvairių pasaulio šalių mokslininkai siekia sukurti sėkmingą balso dialogo sistemą, kuri galėtų interaktyviai bendrauti su žmogumi, siekiant pagreitinti informacijos apdorojimą kompiuteriu ar kitais informacijos apdorojimo įrenginiais.

Šnekos atpažinimas tiriamas apie šešiasdešimt metų, siekiant mechaniškai realizuoti žmogaus šnekos gebėjimą, automatizuoti uždavinius, kuriuos žmogus gali atlikti sąveikaudamas su kompiuteriu [3]. Šnekos atpažinimas *(automatic speech recognition)*, dažnai dar vadinamas

automatiniu šnekos atpažinimu, yra procesas verčiantis žmogaus šneką kompiuteriui suprantama signalų kalba [2]. Pasak Jurafsky, tai kompiuterio programinės įrangos darbo procesas, atliekantis žodžių atitikmenų akustiniams signalams parinkimą [1]. Automatinis šnekos atpažinimas (AŠA) yra sudėtinė balsinių technologijų, apimančių šnekos sintezę, asmens tapatybės vertinimą pagal jo balsą, dalis. Automatinis šnekos atpažinimas yra pirmoji grandis balsinių technologijų produktuose. AŠA technologijų pagrindinis tikslas – sukurti mašinas, kurios galėtų girdėti, suprasti, kalbėti ir veikti pagal gautą informaciją [9].

Remiantis Vaičiukynu, šnekos atpažinimas yra viena iš balso technologijos grupių. Šnekos atpažinimas – balsu tariamų vienetų (frazių, žodžių, jų sekų) automatinis nustatymas t.y., žmogaus šnekos pavertimas tekstu (akustiniams signalams parenkant žodžių atitikmenis) naudojant kompiuterį. (žr. į 1.1 pav.)



1.1. pav. Šnekos atpažinimo sistemos schema (E. Vaičiukynas)

Pagrindinėmis automatinio šnekos atpažinimo sistemos dalimis laikoma: požymių (pvz. kepstrinių) išskyrimas iš šnekos signalo; garsinę informaciją reprezentuojančių modelių (akustinių ir kalbos) formavimas; nežinomo ištarimo klasifikavimas vienam iš reprezentacinių modelių - žodžių ir sakinių atpažinimas. Pagal taikymo sritį ir poreikį, skiriami kalbos atpažinimo sistemų tipai: tęstinės, arba natūralios kalbos – sudėtingesnės formos kalbos programinė įranga, kur vartotojas gali natūraliai paaiškinti problemą, arba prašyti aptarnauti. Tokia sistema pasižymi didele duomenų baze, kur užklausa lyginama su prasminiais žodžiais, ar frazėmis ir parenkamas artimiausias sprendimas, ko vartotojas nori. Kuo aiškesnis tarimas, tuo didesnė tikimybė sulaukti norimo atsako; diskrečios, arba gramatiškai apribotos kalbos – plačiai taikomos klientų aptarnavimo sferoje. Tokia sistema nuo kalbėtojo nepriklausanti ir suprantanti tik ribotą žodžių ar frazių kiekį. Reikalaujama kalbėti aiškiai, lėtai, atskiriant žodžius. Paprastai tokiose sistemose leidžiama pasirinkti tarp “taip” arba “ne” atsakymo;

automatinio kalbos atpažinimo – tokia programinė įranga skiriasi tuo, kad nesistengiama suprasti, kas buvo pasakyta, o tik identifikuoti ištartus žodžius. Klaidų neišvengiama dėl žodžių skambesio panašumo .

Egzistuoja dviejų tipų kalbos atpažinimo režimai kompiuteryje - sinchroninis (*synchronous-single*) ir asinchroninis (*asynchronous – multiple*). Sinchroninis režimas palankesnis greitoms įėjimo/išėjimo operacijoms atlikti, su trumpesniais garsiniais failais, asinchroninis – lėtesnėms, dirbant su didesnės apimties įėjimo signalais. Abu režimai išreiškiami programiškai ir turėtų būti suderinti su kalbos atpažintuvo tipu, nuo ko priklauso sėkminga kalbos atpažinimo operacija. Vis dėl to, šnekos atpažinimas yra labai sudėtingas uždavinys. Prireikė trisdešimties metų, kol atsirado pirmosios praktiškai naudojamos sistemos. Uždavinio sudėtingumą nulemia tokios priežastys:

Keletą kartų ištarto tam tikro garso akustinė realizacija labai skiriasi, net jei jį ištarė tas pats diktorius ir tame pačiame žodyje;

Kalbėjimo greitis gali labai kisti, todėl skiriasi kelių to paties žodžių akustinių realizacijų ilgis. Kintant žodžių ilgiui atskirų garsų ilgis kinta netiesiškai;

Garso akustinė realizacija priklauso nuo gretimų garsų, tai vadinama koartikuliacija;

Kalbėjimo sraute nėra aiškių garsų ar žodžių ribų;

Kiekvieno žmogaus tartis yra skirtinga, todėl reikalingas arba apmokymas konkrečiam diktoriui, arba sistema kūrimo metu turi būti apmokyta su kuo didesniu diktorių skaičiumi;

Jei kuriama atpažinimo sistema remiasi žodžių atpažinimu, žodžių etalonų skaičius gali būti pernelyg didelis;

Kalbėjimo sraute gali būti ir nekalbinių fragmentų (pvz., kosulys), kuriuos reikia atskirti ir pašalinti;

Praktiniuose taikymuose papildomų problemų sukelia foninis triukšmas [7].

Pasak Vaičiūno, siekiant sukurti sėkmingą balso dialogų sistemą, labai intensyviai darbuojasi kalbos atpažinimo specialistai. Plačiai paplitusioms kalboms (anglų, ispanų ir kt.) jau yra sukurtos komercinės atpažinimo sistemos, deja, tokios sistemos lietuviai dar neturi. Lietuvių kalbos atpažinimo tyrimai jau daugelį metų vykdomi Lietuvos mokslo įstaigose, bet yra sukurtos tik eksperimentinės kalbos atpažinimo sistemos. Yra tik keli paskelbti darbai, susiję su rišlios kalbos atpažinimu, bet ir juose pagrindinis dėmesys skiriamas akustiniams atpažinimo modeliams. Labai didelio žodyno rišlios šnekos atpažinimo problema nebuvo nagrinėta ir dėl to, kad tokio atpažinimo sistemoms reikalingas kalbos modelis, o tekstynai, reikalingi tokiems tyrimams, atsirado tik prieš keletą metų [20].

Automatinis šnekos atpažinimas yra taikomas įvairiose žmonių veiklose: klientų aptarnavimas didelėse kompanijose – informacijos teikimas, skambučių priėmimų centrų automatizavimas sumažinant laukimo laiką; automatizuoto teksto rinkimas diktuojant – teksto surinkimas nereikalauja surinkimo įgūdžių ir sunaudojama mažiau žmonių darbo resursų; pasinaudodami automatiniu kalbos vertimu, žmonės gali susikalbėti ir nemokėdami reikiamos kalbos [3]. AŠA gali būti taikomas ir kaip pagalbinė bendravimo priemonė neįgaliems žmonėms, kuriems rinkti tekstą klaviatūra dažniausiai yra sudėtinga, žmonėms turintiems specifinius sugebėjimo skaityti sutrikimus (disleksija), sugebėjimo rašyti sutrikimus (disgrafijų) ar kitokių sunkumų manipuliuojant tekstine forma.

Aktyvūs AŠA tyrimai vykdomi medicinos, farmacijos, telekomunikacijų, multimedijų, automobilių pramonės ir kitose srityse. *“The Telepgraph”* duomenimis, pasaulyje yra daugiau nei 70 milijonų automobilių, kuriuose įdiegta balsu valdomos sistemos (automatinio stabdymo ar statymo technologijos); sukurti balsu valdomi įrenginiai matuojantys kraujo spaudimą, gliukozės ir insulino kiekį kraujyje; išmaniuosiuose telefonuose įdiegti virtualūs balso asistentai, galintys pateikti atsakymus į balsu užduotus klausimus bei vykdantys pateiktas užduotis daugiau kaip 40 kalbų (pvz., navigacijos ar tekstinių žinučių rašymo). Lenkijos bendrovė sukūrė pirmą pasaulyje vien tik balsu valdomą išmanųjį telefoną *„See You“,* skirtą akliesiems ir silpnaregiams naudotis standartinių mobiliųjų telefonų funkcijomis.

Prieš projektuojant šnekos atpažinimo sistemas (plg. angl. *Automatic Speech Recognition,* sutr. ASR) svarbu išsiaiškinti kas jau padaryta šioje srityje, kokios technologijos ir metodai plačiausiai paplitę, kokios galimos praktinės jų pritaikymo problemos. Šiame ir tolimesniuose skyriuose pateikiama šnekos atpažinimo sistemų klasifikacija, veikimo pagrindai, Lietuvos ir užsienio autorių mokslinių darbų šioje srityje analizė. Šnekamosios kalbos atpažinimo sistemų trūkumus paprastai nurodo ne ekspertai. Ar įmanoma atpažinti kas buvo pasakyta triukšmingoje aplinkoje? Kas nutinka, kai šneka užkimęs ar susijaudinęs žmogus. Šie du klausimai apibrėžia du pagrindinius variatyvumo šaltinius ASR sistemose. Išoriniai variatyvumo faktoriai (plg. angl. *extrinsic variables*) priklauso nuo aplinkos: signalas - triukšmas santykis gali būti aukštas, bet kisti laike, skiriasi telekomunikacijų kanalai (laidiniai ar bevieliai), netgi mikrofono pakeitimas kitu modeliu gali smarkiai įtakoti atpažinimo klaidų kiekį. Šnekos signalai perteikia netik semantinę informaciją, bet ir daug informacijos apie patį kalbėtoją: kokia jo lytis, amžius, socialinė ir regioninė kilmė, sveikata, emocinė būsena ir netgi asmens tapatybė. Tai vadinami vidiniai variatyvumo faktoriai (plg. angl.

intrinsic variables) [12]. Įvertinus šių specifinių kintamųjų įtaką galima žymiai pagerinti ASR lankstumą. Reikia pabrėžti, kad šnekos signalas nėra stacionarus. Šnekos spektrinis tankis kinta laike, priklausomai nuo balsaskylės signalo (pvz. įtakoja pagrindinį toną) ir kalbos padargų (liežuvio, lupų ir t.t.) padėties. Pavyzdžiui, toks signalas gali būti modeliuojamas remiantis paslėptais Markovo modeliais, kaip tam tikrų stacionarių atsitiktinių įvykių seka. Pirmoje signalo apdorojimo stadijoje dauguma ASR analizuoja trumpą signalo fragmentą, pagal kurį nustatomas šnekos stacionarumas. Signalo analizei plačiai naudojami įvairūs filtrai, kepstrai, ieškoma specifinių požymių ir pan.

1.2.1. Dinaminis laiko skalės kraipymas

Dinaminis laiko skalės kraipymo metodas, priklausantis pavyzdžių palyginimo metodų grupei, itin išpopuliarėjo praėjusio amžiaus 7 – 8 – jame dešimtmečiuose. Šis metodas buvo dažniausiai taikomas pavieniams žodžiams atpažinti, tačiau buvo bandymų pritaikyti žodžių junginiam ir net ištisinei kalbai atpažinti. Nepaisant to, kad pats metodas ir jo algoritmas sukurtas prieš keturis dešimtmečius, jis sėkmingai taikomas ir šiais laikais.

1.2.2. Paslėptieji Markovo modeliai ir jų pritaikymas kalbos atpažinime

Pasak Henriko Pranevičiaus – atpažįstant šnekamąją kalbą, reikia atlikti du veiksmus: išnagrinėti nedidelį signalo fragmentą ir nustatyti šio fragmento priklausomybę vienai iš galimų klasių arba nustatyti šio fragmento priklausomybę vienai iš galimų klasių, arba nustatyti šio fragmento panašumą į etaloninius signalo fragmentus ir nustatyti visos signalo fragmentų sekos (paprastai vadinamos stebėjimų seka) priklausomybę vienai iš galimų ištarimų klasių. Pirmas uždavinys vadinamas lokaliojo panašumo arba lokaliosios būsenos nustatymo uždaviniu, o antrasis – globaliojo panašumo arba tiesiog atpažinimo uždaviniu. Lokalusis panašumas nustatomas taikant pasirinktą atstumo matą arba nustatant priklausomybės tam tikrai būsenai tikimybę, o globalusis – naudojant dinaminės laiko skalės transformacijos algoritmą arba modeliuojant ištarimą paslėptąja Markovo grandine [11].

Paslėptųjų Markovo modelių (PMM) metodas gali aprašyti tam tikro laike kintančio proceso savybių kitimą ir to kitimo statistines charakteristikas. Pagrindinė paslėptųjų Markovo modelių, naudojamų šnekamajai kalbai atpažinti prielaida yra ta, kad šneka gali būti pakankamai gerai aprašyta kaip atsitiktinis parametrinis procesas ir šio stochastinio proceso parametrai gali būti gana tiksliai nustatyti. Paslėptuosius Markovo modelius ir jų naudojimo šnekamajai kalbai atpažinti galimybes nagrinėjo labai daug tyrėjų [11].

Paslėptųjų Markovo modelių grandines naudoja absoliuti dauguma šiuolaikinių kalbos atpažinimo sistemų. Taigi, remiantis Algimantu Rudžioniu galima išskirti tipiniai šnekamajai kalbai naudojamas PMM struktūras.

Pirmasis modelis vadinamas ergodiniu arba visiškai sujungtu modeliu, nes čia kiekviena būsena gali būti pasiekta iš bet kurios kitos būsenos per vieną žingsnį.

Antroji grandinės tipologija vadinama modeliu „iš kairės į dešinę“ modeliu arba Bakio, nes laike galima judėti tik į kitą arba dar kitą būseną, arba pasilikti toje pačioje būsenoje. Akivaizdu, kad „iš kairės į dešinę“ tipologija atspindi šnekamojoje kalboje vykstančius akustinius procesus: fonetiniai procesai vyksta nuosekliai vienas po kito, o kai kurie fonetiniai įvykiai gali būti praleidžiami dėl garsų asimiliacijos arba tarimo klaidų.

PMM grandinės skirstomos į diskrečiuosius modelius ir tolydžiuosius modelius t.y. – skirstoma pagal tai ar kiekvienai būsenai priskiriami stebimieji įvykiai yra diskretiniai, ar tolydiniai. Tačiau bet kuriuo atveju pats įvykis yra atsitiktinis, todėl pats procesas yra dvigubai stochastinis: viena atsitiktinių įvykių seka yra stebima, tačiau antra, vidinė valdanti stebimąją įvykių seką, yra nematoma ir kartu atsitiktinė. Todėl ir pats modelis vadinamas paslėptosiomis Markovo grandinėmis, nes vidinė dominančioji procesų seka negali būti stebima tiesiogiai, o tik per atsitiktinį išorinį stebimąjį procesą [4]. Konkrečiai susiejant tai, kas pasakyta, su automatiniu šnekamosios kalbos atpažinimu, vertėtų pabrėžti, kad mus domina fonetinių vienetų seka, kuri pati atsitiktinis procesas. Ši seka negali būti stebima tiesiogiai: atpažinimo sistema fiksuoja požymių vektorių sekas, kurios pačios yra atsitiktinis procesas, ir stebimą atsitiktinį procesą turi susieti su tiesiogiai nestebimais

fonetiniais vienetais. Jeigu stebimasis požymių vektorius priskiriamas tam tikrai klasei, o paskui modeliuojama tokių klasių seka, tuomet paslėptoji Markovo grandinė vadinama diskrečiąja, o jei stebimasis požymių vektorius iš anksto nėra priskiriamas jokiai klasei, paliekama patį priskirimą atlikti paslėptajai Markovo grandinei. Diskrečiųjų įvykių PMM aprašoma šiais parametrais:

$O = \{O_1, O_2, ..., O_T, \}$ = stebimųjų įvykių seka (įėjimo seka). T – stebimosios vektorių sekos dydis (vektorių skaičius).

$Q = \{q_1, q_2, ..., q_N\}$ = (paslėptųjų) būsenų skaičius modelyje. M – klasių skaičius arba vektorinės knygos narių skaičius.

$A = \{a_{ij} = P(q_{j\_laiko\_momentu} + 1|)\}$

PMM algoritmai nebūtų tapę tokie populiarūs ir veiksmingi, jeigu nebūtų sukurtas veiksmingas apmokymo algoritmas – Baumo ir Velčo algoritmas. Prie šio algoritmo grįšime kiek vėliau. Paprastai parametrų rinkiniai apskaičiuojami kiekvienam atpažinimo sistemą sudarančiam fonetiniui vienetui (kartais tai gali būti ir pavieniai žodžiai). Atpažinimo metu PMM pateikiamas nežinomas ištarimas, kuriam apskaičiuojama tikimybė (panašumas), kad nagrinėjamoji PMM grandinė sugeneruos tokią stebėjimų seką, kuri atitiks pateiktą nežinomą ištarimą. Tokios tikimybės apskaičiuojamos visoms PMM grandinėms ir išrenkama to ištarimo grandinė, kuriai apskaičiuota tikimybė yra didžiausia. Jeigu atpažinimo sistemos pagrindas paremtas mažesniais negu žodis vienetais, tuomet parenkama PMM grandinių jungtis, kuriai apskaičiuojama suminė didžiausia tikimybė. Grandinei išrinkti naudojamas Viterbio algoritmas: parenkama pora modelio parametrų m^* ir būsenų sekos q^* (m^*, q^*), kurios tenkina sąlygą:

$$(m^*, q^*) = \arg_{(m,q)} \max P(O, q \mid \lambda_m),$$

λ_m – m-oji PMM grandinė (m = 1, 2,…, M);

M – žodyno dydis;

$O = O_1 O_2 ... O_T$ – nežinomą ištarto pjuvių skaičius;

Q – būsenų seka.

Pati tikimybė $P(O, q \mid \lambda_m$ gali būti apskaičiuota naudojant vadinamąjį judėjimo pirmyn ir atgal algoritmą (angl. *Forward – backwardalgorithm*).

Pristačius pagrindinius paslėptųjų Markovo grandinių modelių naudojimo šnekamajai kalbai atpažinti principus galime pereiti prie detalesnio algoritmo nagrinėjimo. Taigi norint sėkmingai naudoti PMM kalbos signalams atpažinti reikia išspręsti tris uždavinius:

Įvertinimo uždavinį: turint stebėjimų seką $O = O_1 O_2 ... O_T$ ir grandinę aprašančio modelio parametrus $\lambda = \{A, B, \pi\}$, reikia apskaičiuoti tikimybę $P(O \mid \lambda)$, kad nagrinėjamoji stebėjimų seka buvo sugeneruota nagrinėjamo modelio;

Paslėptųjų būsenų nustatymo uždavinį: turint stebėjimų seką $O = O_1 O_2 ... O_T$, reikia nustatyti būsenų seką $I = \{i_1, i_2, ..., i_T\}$, kuri būtų optimali tam tikro pasirinkto prasmingo kriterijaus prasme;

Apmokymo uždavinį: kaip parinkti modelio parametrus $\lambda = \{A, B, \lambda\}$, kad būtų maksimizuota tikimybė $P(O|\lambda_M$ [11].

Pagrindinė atpažinimo schema - tolydinio tankio paslėptos Markovo grandinės (continuous density hidden Markov model - CD HMM). Šiuo metu tai yra populiariausia kalbos signalų atpažinimo schema. Bet koks lingvistinis vienetas (žodis, skiemuo, fonema) yra aprašomas tam tikru skaičiumi būsenų ir perėjimo tikimybėmis. Daroma prielaida, kad tai kas ir kaip tariama niekada nėra tiksliai žinoma (paslėptas procesas), bet rezultatą visada stebime (girdime) ir jį galime fiksuoti. Taigi automatinio atpažinimo įtaiso šerdis yra pagal stebėjimo rezultatus sukonstruotas paslėpto proceso modelis. Modelio parametrų įvertinimui naudojama Baum-Welch procedūra, atliekant iteracinius tiesioginių-atbulinių (Forward - Backward) tikimybių skaičiavimus. Sintaksė modeliuojama N-gramatikomis, kuriose yra sukaupiamos N paeiliui einančių žodžių statistikos. Atpažinimo procesas grindžiamas Viterbi algoritmu, kai su ištarta fraze dinaminio programavimo būdu lyginami žinomi CD HMM modeliai, surandant panašiausią [11].

Pašalinių garsų atmetimas. Klausydamas jį dominančios kalbinės informacijos, žmogus sugeba ignoruoti pašalinius pokalbius, triukšmus, muzikinius garsus ar panašiai, žinoma, jei pastarieji nėra pernelyg intensyvūs. Tai reiškia, kad reikia turėti galimybę atmesti įvairius akustinius garsus, kurių nėra kompiuterinio dialogo žodyne. Tinkamai parenkant atpažįstamų signalų panašumo slenkstį, tikrinama ar nagrinėjama komanda yra pakankamai panaši į kurią nors vieną iš leistinų komandų. Jei slenkstis pakankamai aukštas, tai atmetama ir dalis leistinų komandų, o jei šis slenkstis per žemas, atpažinimo įtaisas beprasmiškai reaguoja į pašalinius garsus. Kuo tikslesnis yra atpažinimo algoritmas, tuo efektyviau veikia ši procedūra.

Panašiai, kaip ir bet kurį atpažinimo procesą, kalbos atpažinimą galima išskaidyti į tris pagrindinius etapus: duomenų įvedimą, požymių išskyrimą ir atpažinimą. Kalbos atpažinimo etape nustatoma, kokios fonemos ar žodžiai yra kalbos signale. Tam reikalinga laiko, dažnio ir amplitudinė signalo analizė, kuomet skaičiuojami tam tikrų dažnio juostų energiją charakterizuojantys parametrai [7].

Aštuntąjį praėjusio amžiaus dešimtmetį kalbai atpažinti buvo pradėtas taikyti paslėptųjų Markovo modelių metodas. Pastaraisiais dešimtmečiais tai dažniausiai naudojamas metodas kuriant eksperimentines ir komercines automatinio kalbos atpažinimo sistemas. Tolydinio tankio paslėptos Markovo grandinės (*continuous density hidden Markov model - CD* HMM), šiuo metu yra populiariausia kalbos signalų atpažinimo schema. Bet koks lingvistinis vienetas (žodis, skiemuo, fonema) yra aprašomas tam tikru skaičiumi būsenų ir perėjimo tikimybėmis. Daroma prielaida, kad tai kas ir kaip tariama, niekada nėra tiksliai žinoma (paslėptas procesas), bet rezultatą visada stebime (girdime) ir jį galime fiksuoti. Taigi automatinio atpažinimo įtaiso šerdis yra pagal stebėjimo rezultatus sukonstruotas paslėpto proceso modelis [10].

Paslėptųjų Markovo modelių (PMM) taikymo metodika remiasi tikimybiniu modelių sudarymu. Tai statistiniais pavyzdžiais pagrįstas atpažinimo metodas. Šiuo būdu atliekama spektrinė kalbos signalo analizė, kurios metu gaunami požymio vektoriai, kurie aprašo įvairius kalbos garsus. Šie garsų modeliai sujungiami į tinklą, atsižvelgiant į jų tvarką įvairiuose žodžiuose.

PMM naudojami bioinformatikos, automatinio teksto, muzikos generavimo, kalbos atpažinimo, lošimų ir daugelyje kitų sričių. Dėl šio modelio paprastumo jį galima integruoti į įvairius naujus modelius, taip pat įvairiai modifikuoti, priklausomai nuo konkretaus uždavinio.

PMM modeliuoja atsitiktinius procesus. Atsitiktinis procesas juda būsenų seka, kiekvienoje būsenoje generuodamas kokį nors įvykį. Skiriami:

Stebimas Markovo modelis. Jis pasižymi tuo, kad pagal įvykių seką galima surasti būsenų seką. Čia viena būsena atitinka vieną įvykį.

Paslėptasis Markovo modelis. Pagal įvykiu seką negalima atstatyti būsenų sekos. Čia įvykis yra tikimybinė būsenos funkcija, nes viena būsena atitinka kelis įvykius, pasikartojančius kitose būsenose.

Daugeliui procesų modeliuoti dėl jų sudėtingumo yra taikomas paslėptasis Markovo modelis. Jis tinka ir šnekos atpažinimo modeliavimui, nes šnekos signalas yra laike kintantis stochastinis (atsitiktinis) procesas. Paslėptasis Markovo modelis [17]. apibrėžiamas kaip baigtinė būsenų, susietų perėjimo tikimybėmis, seka, naudojama signalų laikiniam ir spektriniam kitimui modeliuoti. Paslėptas Markovo modelis aprašomas nusakant penkis dydžius:

Būsenų Q = (q1, q2,…,qN) skaičių N.

Skirtingų stebėjimų būsenoje skaičių D – diskrečiu atveju arba stebėjimų $\mathbf{O} = \mathbf{o}_1,\mathbf{o}_2,\ldots,\mathbf{o}_T$ paskirstymo tankį – tolydžiu atveju.

Perėjimo iš vienos būsenos į kitą tikimybių matricą $\mathbf{A} = \{a_{ij}\}$,

$i,j = 1,\ldots,N$, $\forall i,j \quad a_{ij} \geq 0$ ir $\forall i \sum_{j=1}^{N} a_{ij} = 1$.

Stebėjimų tikimybinius skirstinius $\mathbf{B} = \{b_j(k)\}$, $k = 1,\ldots,D$, $j = 1,\ldots,N$

- diskrečiu atveju arba stebėjimų tikimybės tankio funkcijas

$\mathbf{B} = \{b_j(\mathbf{o}_1)\}$, $j = 1,\ldots,N$, $t = 1,\ldots,T$ – tolydžiu atveju.

Pradinio (initial) buvimo būsenoje tikimybes $\pi_i = P(q_1 = i)$, i = 1,…,N. Šnekos atpažinimo modeliavime pradine būsena laikoma pirmoji paslėptojo Markovo modelio būsena $\pi_i = 1$.

Šie penki dydžiai nusako konkretų paslėptąjį Markovo modelį, nors tam užtenka nusakyti rinkinį $\lambda = (\mathbf{A},\mathbf{B},\pi)$. Kadangi π yra konstanta visiems PMM, naudojamiems šnekai atpažinti, rinkinys paprastėja iki $\lambda = (\mathbf{A},\mathbf{B})$ [9].

Vienas iš pagrindinių veiksnių, lėmusių PMM populiarumą sprendžiant šnekos atpažinimo uždavinius, yra mokymosi algoritmų egzistavimas). Mokymui naudojamas iteracinis Baum-Welch algoritmas su kuriuo turint garsyną, galima surasti modelio parametrus A ir B. (Vaičiūnas A., 2006:24) Baum - Welch procedūra naudojama Markovo modelio parametrų įvertinimui, atliekant iteracinius tiesioginių-atbulinių (*Forward - Backward*) tikimybių skaičiavimus. Sintaksė modeliuojama N-gramatikomis, kuriose yra sukaupiamos N paeiliui einančių žodžių statistikos. Atpažinimo procesas grindžiamas Viterbi algoritmu, kai su ištarta fraze dinaminio programavimo būdu lyginami žinomi CD HMM modeliai, surandant panašiausią [10]. Kitaip tariant, Viterbi algoritmas suranda labiausiai tikėtinus kelius Paslėptuose Markovo modeliuose ir išrenka PMM su didžiausia tikimybe.

Taigi, atpažinimo metu daroma prielaida, kad nežinomą ištarimą atitinkanti tiriamų požymių vektorių seka yra gaunama-generuojama PMM sekos. Skaičiuojamos to ištarimo atitikimo akustinių modelių deriniams tikimybės ir labiausiai tikėtina modelių kombinacija identifikuoja ištarimą [9].

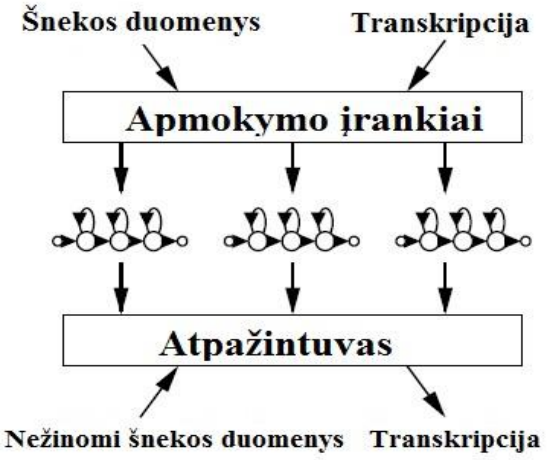
1.2.3. Dirbtiniai neuronų tinklai

Dirbtiniai neuronų tinklai, kalbai atpažinti pradėti taikyti praėjusio amžiaus devintajame dešimtmetyje, yra jauniausias iš nagrinėjamų metodų. Iš pradžių neuronų tinklai taikyti fonemoms [12], skiemenims atpažinti, vėliau atskiriems žodžiams ir ištisinei kalbai. Tačiau reikėtų pasakyti, jog neuronų tinklai nepasiteisino kaip savarankiškas atpažinimo metodas (ypač ištisinės kalbos atpažinime) ir dažnai naudojami kartu su paslėptaisiais Markovo modeliais.

Pagrindinis visų dirbtinių neuronų tinklų elementas – neuronas – supaprastintas biologinio neurono modelis, sudarytas iš branduolio su įėjimo ir išėjimo taškais.

1.3. HTK paketas

HTK atpažinimo programų rinkinys yra vienas populiariausių ir plačiausiai naudojamų atpažinimo įrankių. HTK pirmiausiai yra skirtas šnekos apdorojimo priemonių, paremtų paslėptaisiais Markovo modeliais, realizavimui. Yra du pagrindiniai apdorojimo etapai: pirmiausiai, HTK mokymo įrankiai yra naudojami siekiant apytikriai apskaičiuoti paslėptųjų Markovo modelių rinkinio parametrus, naudojant mokymui pateiktus pasakymus (įrašus) ir su jais susijusias transkripcijas; antra, nežinomi pasakymai/įrašai yra transkribuojami naudojant HTK atpažinimo įrankius [15]. Žemiau pateikiama schema, vizualiai apibūdinanti šiuos du svarbiausius apdorojimo etapus. (žr 1.1. pav.)



1.2. pav. HTK atpažinimo įrankio šnekos apdorojimo schema

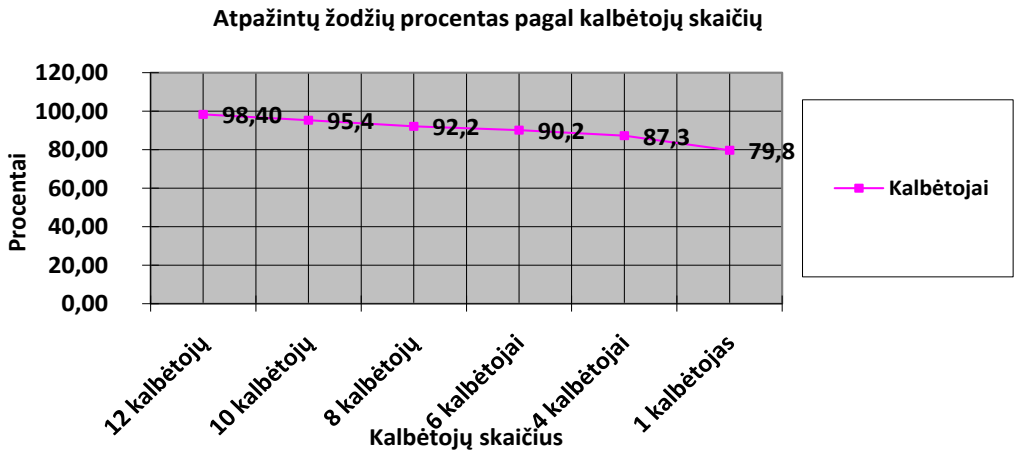
Vykdant atpažinimą su HTK paketu, kiekvienam etapui atlikti reikia panaudoti po atskirą programą iš HTK atpažinimo programų rinkinio. Visos programos yra valdomos iš komandinės eilutės per papildomus failus ir eilutės komandas. Žemiau pateikiami pagrindiniai darbo etapai ir jų paaiškinimai su šiuo paketu bei jų įgyvendinimui naudojamų programų pavadinimai:

Žodyno sudarymas. Kad projektuojama sistema žinotų, kokios žodžių (akustinių vienetų) kombinacijos yra leidžiamos ir kokios ne;

Požymių apskaičiavimas. Atpažinimo sistemose naudojami ne patys įrašai, o iš jų apskaičiuoti požymiai, vadinasi norint apskaičiuoti požymius pirmiausiai reikia turėti įrašų failus. Požymių skaičiavimui naudojama programa Hcopy;

Modelių apmokymas. Apmokymas vyksta naudojant Baum-Welch algoritmą, paleidus programą pavadinimu HRest. Svarbus apmokymo etapas – modelių failų kūrimas, kuriuose nurodomi pagrindiniai HMM modelio parametrai. Modelių failai – tai specialaus formato failai, kuriuose bus surašoma atitinkamo modelio parametrai: perėjimų tikimybės ir išėjimų tikimybės. Kadangi HTK operuoja tik tolydiniais HMM modeliais, tai ir išėjimų tikimybės modeliuojamos Gauso skirstiniais. Vadinasi išėjimų tikimybės aprašomos vidurkiais ir standartiniais nuokrypiais;

Atpažinimas. Atlikus apmokymą ir turint testavimui skirtus įrašus galima atlikti testavimą ir patikrinti kaip gerai vykdomas atpažinimas. Atpažinimui patartina naudoti kitus įrašus negu buvo naudoti apmokymui. Testavimas vykdomas paleidus programą HVite.



1.3. pav. Atpažintų žodžių procentų kitimas priklausomai nuo diktorių skaičiaus

Viršuje pavaizduotoje diagramoje (žr. 1.3. pav.), matome kaip kinta atpažinimo rezultatai esant daugiau kalbėtojų. Aiškiai matoma, jog kuo daugiau diktorių pasako tą patį žodį, tuo geresni rezultatai.

1.4. Garsynai

Šiandien kalbos technologijų tyrimai neįsivaizduojami be garsynų. Įvairiuose žodynuose jie yra apibūdinami kaip kalbos garsų sistema, jų visuma, vokalizavimas. ***Garsynas*** *(phonetic database)* – tai konkrečiai kalbai surinkta struktūralizuota garso įrašų aibė, kurioje kiekvienas garso įrašas turi atitikmenį tekstu (transkripciją fonemomis, skiemenimis, žodžiais), žodyną. Užsienio literatūroje garsynai apibrėžiami kaip specialiai surinktų kalbos signalų duomenų bazės (*angl. speech corpora*). Pasak Balvočiaus, garsynais priimta laikyti šnekos signalų rinkinius, skirtus šnekos atpažinimo, sintezės, diktoriaus identifikavimo, verifikavimo ar kitų uždavinių, reikalaujančių šnekos signalo apdorojimo, sprendimui [6]. Laboratorijose vykdomi kalbos atpažinimo sistemų modeliavimai būtų neįmanomi, jei nebūtų garsynų, pavadintų „didelės apimties šnekos signalų imtimis“. (Laurinčiukaitė

S., 2008) Bendriausiai garsynai gali būti apibūdinami kaip garsų fondai ar garsų saugyklos, kurių paskirtis reprezentuoti kalbą, pasitarnauti jos tyrimuose. Kalbos išsamiems tyrimams atlikti, kuriami sisteminiai, gerai anotuoti garsų fondai [15].

Kaip teigia Balvočiaus (2003), garsynai skiriasi trukme, transkribavimo lygmeniu, diktorių skaičiumi, žodyno dydžiu, renkamų garso įrašų turiniu ar pan. Atpažinimui naudojant tam tikrus garsynus didelę įtaką daro ir triukšmai bei kiti trukdžiai esantys garsynuose [6]. Pagal paskirtį garsynus būtų galima išskirti į tokias grupes: garsynai, kurie buvo kurti balso atpažinimo nuo diktoriaus nepriklausančias sistemas ir yra sistemos, kurios balsą atpažįsta, tačiau turi būti pritaikomos (apmokamos) prie diktoriaus. Pagal tipą garsynai skirstomi į garsynus, kuriuose surinkti įrašai yra "perskaitytos šnekos" (kitaip sakant nespontaniškos šnekos įrašai) (pvz.: knygų ištraukos, tam tikri žodžių sąrašai, skaičių sekos ir pan.) ir garsynus, kurių turinį sudaro spontaniškos kalbos įrašai (pvz.: dialogai ir kt.). Garsynų anotavimui ir segmentavimui įprastai naudojami tie patys metodai, kaip ir šnekos atpažinimui: neuroniniai tinklai, DLSK (dinaminis laiko skalės kraipymas, angl. - DTW – Dynamic Time Warping) ir HMM modelis. Dažniausiai šie metodai taikomi šnekos technologijų tinklalapiuose aprašytose anotavimo ir segmentavimo sistemose [19].

Klasikinio garsyno pavyzdžiu laikomas TIMIT (Texas Instruments/Massachusetts institute of Technology) šnekos duomenų bazė, kurios esmė – direktorijų struktūroje užkoduota pagrindinė garsyno organizacinė informacija, o tekstiniuose failuose lentelėse – papildoma informacija apie šnekos signalus, diktorius, jų anotacijos. TIMIT duomenų bazės duomenys sutalpinti į keturių tipų failus, kurie saugo informaciją tiek apie kalbėtojus, tiek ir anotacijas [6]. TIMIT duomenų bazės struktūra tapo prototipu, į kurį orientuojamasi kuriant naujus garsynus. Ši bazė unikali tuo, kad turi labai kruopščiai sužymėtas fonemų ribas. Daugelyje kitų bazių pateikiamos tik žodžių ribos. Būtent todėl ji tapo plačiai naudojama ir palaipsniui, adaptavus įvairiems ryšio kanalams, buvo transformuota į kitas (NTIMIT, CTIMIT, FFMTIMIT, HTIMIT). TIMIT garsyno pavyzdžiu buvo sudarytas lietuvių kalbos signalų (LTDIGITS) garsynas. LTDIGITS garsynas, sudarytas KTU ir VU.

Lietuvoje žmonių grupės, užsiimančios vien lietuvių šnekos garsynų kūrimu, nėra. Tuo tenka rūpintis patiems šnekos tyrėjams. Šiuo metu garsynus renka ir ruošia Matematikos ir informatikos institutas (MII), Vytauto Didžiojo (VDU), Kauno technologijos (KTU) ir Vilniaus (VU) universitetai. Garsyną kaip produktą apsprendžia jį ruošianti žmonių grupė. Svarbu, kad į grupę patektų kuo įvairesnės specializacijos žmonių – lingvistų, programuotojų, vartotojų. Tada galima tikėtis, kad garsynas atspindės vartotojų poreikius, atitiks galiojančias kalbos normas ir bus aprūpintas programine įranga, leidžiančia lanksčiai dirbti su garsyne esančiais duomenimis [9]. Sukurti universalų garsyną yra sudėtingas uždavinys, nes toks garsynas turėtų gerai aprašyti kalboje sutinkamų fonetinių vienetų įvairovę, įvertinti kontekstinius efektus, diktorių ir kalbėjimo stilių įvairovę ir pan. Todėl garsynai sudaromi orientuojantis į tam tikros uždavinių klasės sprendimą [5]. Raškinis teigia, kad „bet kokių kalbos inžinerijos projektų – taikomųjų ar tiriamųjų – sėkmės pirmiausiai priklauso nuo to, kokiais kalbos duomenų ištekliais (tekstynais ir garsynais) disponuoja tas projektas" [13]. Sistemiškumas, t.y. duomenų struktūrizavimas, jų patalpinimo būdas bei papildomos informacijos žymės garsynuose itin svarbus kriterijus apskritai, nuo kurio priklauso garsyno informatyvumas [19].

Remiantis populiariausių garsynų analizės duomenimis nustatyta, jog nėra visus reikalavimus tenkinančio garsyno. Išeitis – siūlyti naujas duomenų modelių ir programinės įrangos architektūros kombinacijas [15]. Pasak Balvočiaus ir Telksnio, šiuolaikinis garsynas, kuris galėtų būti efektyviai naudojamas šnekos atpažinimo sistemų kūrimui, turi tenkinti sekančias savybes:

- suteikti vartotojui galimybes manipuliuoti dideliais duomenų kiekiais (perkelti, atnaujinti, naikinti) ir užtikrinti duomenų korektiškumą bei integralumą;
- turėti efektyvią paieškos sistemą, užtikrinančią tiek šnekos signalų, tiek meta-duomenų ištraukimą;
- turėti savybes, atitinkančias bendro darbo sistemos principus (daugiavartotojiškumas, duomenų kontrolė ir dalinimasis);
- turėti standartinę sąsają duomenų lygyje (duomenų importavimui ir eksportavimui į kitas sistemas);

- turėti standartinę programinę sąsają, kad sukūrus papildomą funkcionalumą, jis taptų lengvai prieinamas ir kitiems vartotojams;
- turėti vartotojo sąsają patogiam duomenų bazės užklausimui ir ataskaitų formavimui [6].

Šnekos duomenų bazių, arba garsynų kūrimas prasidėjo, kai pradėta spręsti su šnekos apdorojimu susijusias problemas. Pirmuoju garsynu, kuris buvo ir yra naudojamas šnekos sistemų vertinimui – laikomas TI-DIGITS. Jis surinktas ir publikuotas 1984 metais. Panašių garsynų poreikis labai išaugo smarkiai padidėjus kompiuterių skaičiuojamajai galiai ir kai tapo įmanoma spręsti taikomuosius uždavinius šnekos atpažinimo srityje. Priėjus susitarimo, kad reikalingi garsynai, kuriuos tyrinėtojai galėtų tarpusavyje dalintis, imta kurti organizacijas, prižiūrinčias ir patariančias garsynų rinkimo klausimais. Svarbesnės organizacijos yra Lingvistinių duomenų konsorciumas – LDC (Linguistic Data Consortium) JAV, Europos Kalbinių resursų asociacija (European Language Resources Association – ELRA). Šios organizacijos koordinuoja kalbinių resursų rinkėjų veiksmus, apibrėžia standartus, padeda platinti garsynus, tekstynus ir kitus kalbos resursus [6].

Taigi, garsynų, dar vadinamų *šnekos duomenų bazėmis,* kūrimas prasidėjo, kai pradėta spręsti su šnekos apdorojimu susijusias problemas ir didėja plečiantis taikomajai informatikos, t.y. kalbų technologijų sričiai, todėl šiomis dienomis į kalbos atpažinimo sistemų, tuo pačiu – taikomųjų garsynų kūrimą bei vystymą orientuotos ne tik mokslo, bet ir verslo, prekių, paslaugų organizacijos: auga taikomosios paskirties garsynų apimtys bei finansavimas, kur „kalbos atpažinimo ekonominė nauda geriausiai atsiskleidžia sukuriant naujas telekomunikacines paslaugas [5]. Balso serveriai (Microsoft Speech Server 2007)

Microsoft Balso Serveris 2007 yra balso atpažinimo platforma, kuri yra tiesiogiai susieta su *MS Visual Studio 2005* ir taip susijungia į interaktyvų balso atsakiklį (interactive voice responce – *IVR*), pritaikant VoIP technologiją (*voice over IP*), žiniatinklio ir telefoninio ryšio ypatumus. Ši bendra sistema vadinama Microsoft OCS 2007 (Office Communication Server).Jis yra skirtas atlikti kasdienes paslaugas telefonu – banko sąskaitų tikrinimas, automatinis gedimo registravimas ir kt.
Balso serverio aplinka leidžia atlikti tokias funkcijas: balsu valdyti kompiuterio komandas; *touch-tone*(elektroninis valdymas klavišais) funkcija leidžia valdyti pokalbio eigą spaudžiant telefono klavišus; *text-to-speech* funkcija paverčia klaviatūros įvesties komandas kalba, taip perteikiant jį vartotojams.

Naujojoje Balso serverio 2007 versijoje atsiranda galimybė atlikti internetinius skambučius pritaikant VoIP technologiją. Be to, naujoji versija palaiko tris projektų tipus:

1. SALT (Speech Application Language Tags) – internetiniam ir telefoniniam bendravimui pritaikyti W3C kalbos standartai.

2. VoiceXML, kuris apibūdina interneto, telefono kalbinių priemonių integracija, todėl VoiceXML pritaikytą IVR glima lengvai perkoduoti balso serveriui neprarandant informacijos.

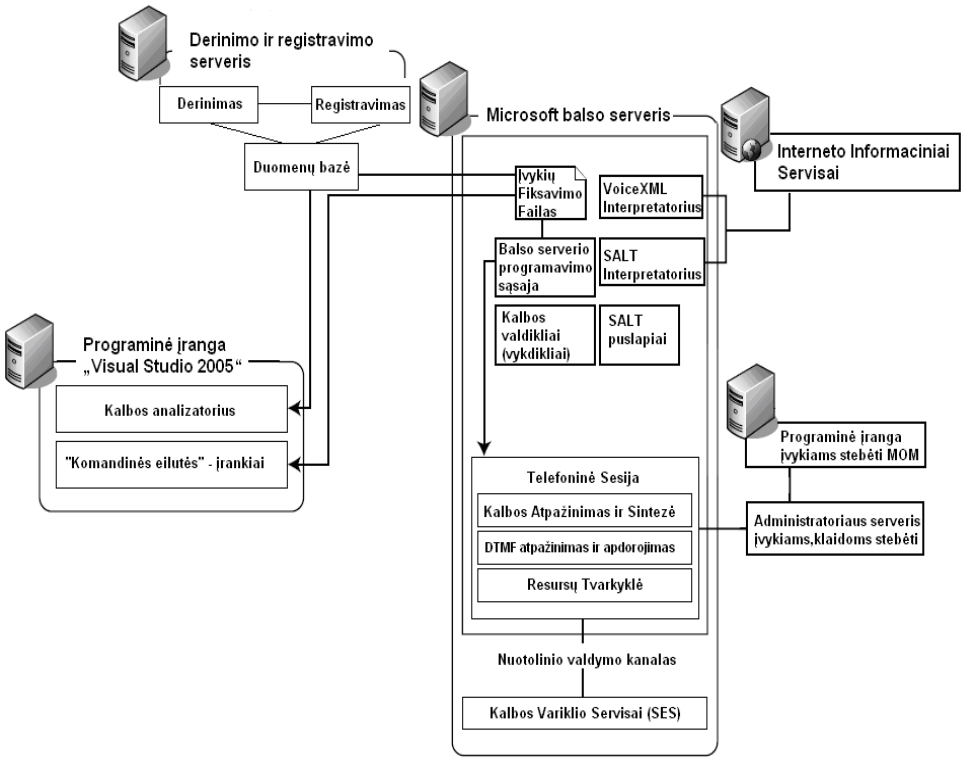
3. Voice Responce Workflow suteikia galimybę stebėti operacijų veikimo eigą, ko iki šiol neleido daryti nei SALT nei VoiceXML.

Prie viso to, pridėta ir naujų valdymo įrankių:

1. Šnekos gramatikos sudarymo įrankis ir gramatikos dizaino patarėjas – šie įrankiai padeda greitai ir lengvai sudaryti gramatikas natūralios kalbos eiga. Yra galimybė sudaryti gramatikas pačiam arba sekant *MS Visual Studio 2005* gramatikos sudarymo įrankio instrukcijomis. Patarėjas padeda nustatyti klaidas, jas ištaisyti.

2. Žodyno redaktoriaus įrankis – pradeda sudaryti arba keisti žodžių tarimą, skirtą šnekos gramatikos sudarymo įrankiui. Teisingai panaudojus šį įrankį, jis gali nulemti, kaip tiksliai bus atpažįstama kalba.

3. Tarties redaktorius – jį naudojant galima redaguoti bei pildyti žodžių tarimus, bet jis veikia tik su *MS Visual Studion 2007* gramatikos sudarymo įrankiu (žr.1.4. pav.).

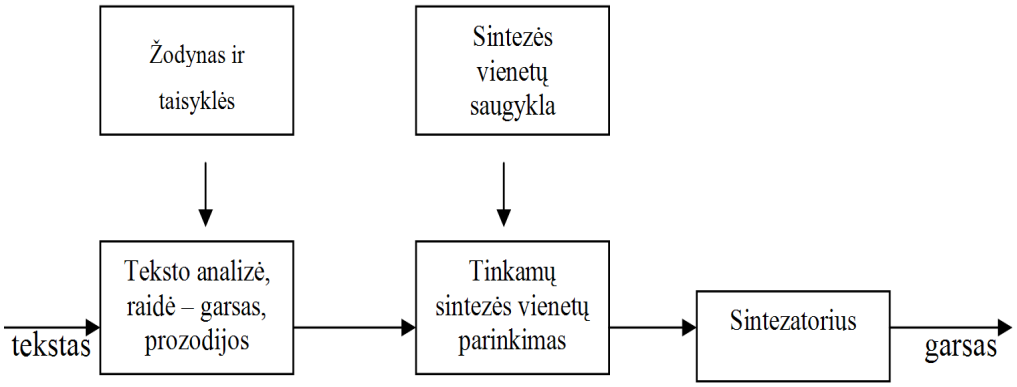


1.4. pav. Balso serverio struktūra

1.5. Kalbos sintezavimas

Kalbos sinteze vadinamas automatinis balsinio pranešimo generavimas iš pateikto teksto ar kitos simbolių sekos, t.y. tekstu pateiktos informacijos skaitymas balsu. Nėra abejonių, kad sintezė - labai nuo konkrečios kalbos savybių priklausanti kalbos technologijų sritis. Generavimui reikia naudoti konkrečiai kalbai paruoštus sintezės elementus (pastarieji dažnai vadinami sintezės vienetais) bei atsižvelgti į duotos kalbos gramatines ypatybes (kirčiavimą, prozodines, intonacines savybes ir pan.). Reikėtų pažymėti, kad lietuvių kalba nėra sintezės požiūriu lengva kalba [10]. Kalbos sintezės pagal tekstą sistemas patogiausia apibrėžti kaip sistemas, automatiškai generuojančias žmogaus balsą naudojant grafemų keitimą fonemomis [8]. Sintezės pagalba pagal reikiamą komandą balsu perskaitoma informacinėje sistemoje teksto pavidalu saugoma informacija. Sintezei priskiriamos ir paprastesnės informacijos pateikimo balsu formos, pvz. iš anksto paruoštu žodžių ar jų sekų pateikimas balsu, esant tam tikram reikalavimui [14].

Kalbos sintezę pagal tekstą pirmiausia patogu išskaidyti į du pagrindinius etapus: lingvistinį teksto apdorojimą ir kalbos signalo formavimą. Taigi bendra kalbos sintezės pagal tekstą funkcinė diagrama atrodo taip, kaip pavaizduota 2 pav.



1.5. pav. Kalbos sintezės schema

Lingvistinio apdorojimo blokas pagal įvedamą tekstą sukuria jo fonetinę transkripciją ir reikiamą intonaciją bei ritmą (dar vadinamus prozodija). Kalbos signalo formavimo blokas gaunamą simbolinę informaciją paverčia į žmogaus kalbą [7].

Lingvistinio teksto apdorojimo metu greta trijų pagrindinių uždavinių (transkribavimo, intonacijos ir garsų trukmių modeliavimo) dar gali būti atliekama visa aibė papildomų teksto apdorojimo darbų: pradinis teksto apdorojimas, morfologinė, kontekstinė bei sintaksinė analizė, žodžio skiemenavimas bei kirčiavimas, sakinio skaidymas į frazes, frazės kirčio radimas ir kt. [8]. Rudžionis patvirtina, jog norint tekstą paversti balsu, reikia nuosekliai atlikti eilę procedūrų. Visų pirma tekstą reikia tinkamai paruošti, atlikti jo transkripciją (pvz. lietuvišką žodį *gąsdina* mes tariame *gazdina*), nustatyti frazėje esančių žodžių kirčius, apibūdinti frazės tipą (paprasta, klausiamoji ar šaukiamoji intonacija). Šis etapas paprastai vadinamas teksto normalizavimu. Galiausiai tekstas paverčiamas fonemų seka su prozodijomis. Šią seką reikia paversti tinkama sintezės vienetų seka, o pastarąją paversti balsu [5].

Remiantis Rudžioniu (2001), sintetinės kalbos kokybei apibūdinti naudojama aiškumo ir natūralumo sąvokos. Aiškumas nusako, kokia dalis lingvistinių vienetų yra suprantama klausytojui. Natūralumas nusako, kiek sintezuota kalba yra artima žmogaus kalbai. Šiuo metu formuojasi nauji vertinimo rodikliai – sintetinės kalbos gradacija pagal vartojimo ir natūralią kokybę. Vartojimo kokybės sintetine kalba laikoma kalba, kai didelė vartotojų dalis moka už paslaugas, kai informacija pateikiama sintetine kalba. Natūrali kokybė suprantama kai klausytojas nesugeba atskirti sintetinės kalba nuo natūralios. Balso technologijų įvertinimui naudojama penkiabalė sistema, kadangi natūralumas yra subjektyvus kriterijus. Skirtingų klausytojų vertinimo rezultatai suvidurkinami.

Anot Kasparaičio (2005), vienas iš labai svarbių testams keliamų reikalavimų yra rezultatų pakartojamumas, t. y. skirtingu laiku skirtingose vietose ir su skirtingais žmonėmis atliekant testą turi būti gaunami panašūs rezultatai. Testo rezultatai priklauso nuo tokių 5 pagrindinių veiksnių:

1. sintezuotos kalbos kokybė;
2. kalbos fragmentų dydis ir sudėtingumas;
3. klausytojo trumpalaikės atminties galimybės;
4. klausymosi ir kitų lygiagrečiai atliekamų užduočių sudėtingumas;
5. klausytojo patirtis klausantis šio ar panašaus sintezatoriaus sintezuotos kalbos. (Kasparaitis P., 2005:11)

Aparatūrinė ar programinė kompiuterinė sistema, galinti žmogaus balsu perskaityti bet kokį jam pateiktą tekstą vadinama kalbos sintezatoriumi, nesvarbu, ar šį tekstą įvedė operatorius klaviatūra, ar jis buvo įvestas naudojant kokią nors rašytinio teksto atpažinimo (angl. optical character recognition - OCR) sistemą [7]. Esminis skirtumas tarp bet kokio kalbančio įrenginio, pavyzdžiui, kasetinio magnetofono, ir sintezatoriaus yra tai, kad sintezatorius gali generuoti naujus sakinius. Lietuvių kalbos sintezatoriai

Pasaulyje sukurta daug sintezatorių, kalbančių daugeliu pasaulio kalbų. Pvz., *Digital Equipment Corporation* sintezatorius *DEC talk* kalba anglų, vokiečių, prancūzų ir ispanų kalbomis, Telia

Promotor AB sintezatorius Infovox 230 kalba anglų, danų, suomių, prancūzų, vokiečių, islandų, italų, norvegų, ispanų, švedų ir olandų kalbomis. Kompanija *Dolphin Systems for People with Disabilities* yra sukūrusi sintezatorių Apollo II, kuris, greta kitų kalbų, kalba ir lietuviškai [Dolphin Speech Synthesizer Series 2 User Guide] [8].

1995 m. VU profesorius P. Kasparaitis sukūrė antros kartos sintezatorių "Aistis", kurio fonetinių vienetų bazėje yra 476 elementai. Sintezatoriuje naudojamas diktoriaus J. Šalkausko balsas. Fonetinių vienetų bazę sudarė prof. A. Girdenis. Naudojami įvairių ilgių fonetiniai vienetai: atskirų garsų dalys (pvz.., sprogstamieji priebalsiai sudaromi iš dviejų dalių), atskiri garsai (balsiai, priebalsiai), garsų poros (dvibalsiai, mišrieji dvigarsiai). Remiantis [7]. sintezatorius sudarytas iš 4 blokų:

- žodžių skiemenavimo, kuriam naudojamas lietuvių kalbos skiemens struktūros algoritmas, priešdėlių atskyrimas bei balsių kombinacijos, kurios negali priklausyti vienam skiemeniui;
- žodžių kirčiavimo, kuriame žodžiai suskirstyti į 3 grupes (daiktavardžiai ir būdvardžiai, veiksmažodžiai, nekaitomi žodžiai), kurioms sukurtas atskiras kirčiavimo algoritmas;
- transkribavimo, kuriame naudojama 700 formalių taisyklių;
- šnekos signalo formavimo, kuriam naudojamas konkatenacinis metodas, kuriame jungiami natūralios diktoriaus kalbos segmentai.

2006 m. pabaigoje kompanija „Rosasoft" kartu su P. Kasparaičiu patobulino anksčiau sukurtą sintezatorių „Aistis". Šiam sintezatoriui suteiktas vardas "Aistis 2", kurį vartotojai pradėjo naudoti 2007 m. pradžioje.

1996 m. Belgijos TCTS laboratorija priklausanti Mons politechnikos fakultetui (*Faculte Poletechnique de Mons*) sukūrė projektą MBROLA. MBROLA – kalbos sintezės sistema, paremta dvigarsių suliejimo sintezės strategija. Iki šiol „MBROLA" buvo pritaikyta 26 pasaulio kalbų sintezei. „MBROLA" sintezė yra paremta TDPSOLA algoritmu, kuris vykdo fonetinių vienetų sklandų sujungimą, ilgio ir pagrindinio tono koregavimą. Lietuvių šnekos sintezei panaudota 1321 dvigarsių segmentų iš VDU bendrinės lietuvių šnekos anotuoto garsyno. Skiriamos 2 MBROLA kalbos sintezavimo strategijos:

Balso trakto modeliavimu paremta sintezės strategija. Šio tipo sistemos skaitmeniškai imituoja kalbos trakto veikimą, todėl šis sintezės modelis labiausia yra priimtinas fonetikos ir fonologijos specialistams.

Garsų suliejimu paremta sintezės strategija. Šis modelis sintezuoja, sujungdamas iš garsų bazės parinktus atitinkamus garsus [12].

2008 m. pavasarį socialinė (įdarbinusi neregius) informacijos technologijų bendrovė „Etalinkas", vadovaujama E. Biknevičiaus, panaudodama Europos struktūrinių fondų paramą – 600 tūkst. litų, sukūrė naują lietuvišką sintezatorių „Sakrament LIT", veikiantį ne tik „Windows", bet ir „Linux" terpėse, kuris visą kompiuterio ekrane pateikiamą informaciją perskaito balsu. Programoje panaudotas buvusio sporto komentatoriaus Vasilijaus Kuzminsko balsas. Etalink duomenimis, sintezatorius naudingas ne tik kompiuterių, tačiau ir mobilaus ryšio įrangos bei kitų prietaisų vartotojams. Naujasis kalbos sintezatorius įgarsintas tikru balsu ir suderinamas su šiuo metu naudojamomis kompiuterio ekrano informacijos skaitymo programomis. Gausesnis garsinės informacijos prieigos priemonių pasirinkimas atveria didesnes galimybes informacijos technologijų srityje ne tik regėjimo negalią turintiems žmonėms, tačiau ir apskritai visiems darbdaviams. Darbo vietos regėjimo negalią turinčiam darbuotojui įrengimas, naudojant ekrano skaitymo ir kalbos sintezavimo priemones, yra kelis kartus pigesnis negu naudojant priemones, kurioms reikalingas Brailio raštas.

Apibendrinant, sintezės iš teksto panaudojimo perspektyvos labai plačios - jos gali būti naudojamos įvairiausiose informacinėse sistemose (ryšiuose, transporte, gal būt sveikatos apsaugoje). Pasaulyje jau egzistuoja visa eilė kalbos sintezės taikymo praktikoje pavyzdžių, dažniausiai orientuotų į didžiąsias pasaulio kalbas (anglų, kinų, prancūzų, vokiečių, japonų). Jose naudojama pakankamai aukštos kokybės balso sintezė, tačiau net ir geriausių šiuolaikinių sintezės sistemų generuoto balso kokybė gerokai nusileidžia natūraliam balsui [5]. Lietuvoje dar nėra sukurtų

trečios kartos sintezatorių, o jau turimi antros kartos sintezatoriai yra prastos kokybės ir praktikoje tinkami tik neįgaliesiems.

1.5.1. Vieneto išrinkimo sintezė

Dominuojantis sintezės iš teksto metodas yra vieneto išrinkimo sintezė (*Unit Selection Synthesis USS*). Antros kartos sintezės iš teksto sistemose vyravo difoninė konkatenacinė sintezė. Ji rėmėsi dviem prielaidom:

1. To paties difono variacijos iššauktos pagrindinio tono ir trukmės pokyčių;
2. Signalo apdorojimo algoritmai gali atlikti difonų pagrindinio tono ir trukmių modifikavimą nepablogindami difonų kokybės.

Gana greitai paaiškėjo, kad pirmoji prielaida nėra teisinga. USS idėja – kiekvienam baziniam lingvistiniam tipui turime visą eilę vienetų (*units*), besiskiriančių prozodinėmis ir kitomis charakteristikomis. Sintezės metu algoritmas parenka (*selects*) vieną vienetą iš galimų pasirinkimų, t.y., vietoje vieno difono (antros kartos difoninė sintezė) turime daug galimų pasirinkimų (trečios kartos USS). Naudojamos didelės kalbos įrašų bibliotekos (daugiau nei valandos trukmės kalbos įrašų archyvai). Tokioje duomenų bazėje kiekvienas įrašas segmentuotas į garsus, skiemenis, morfemas, žodžius, frazes ir sakinius. Paprastai segmentavimas vykdomas analizuojant garso spektrogramą. Nurodomi tokie parametrai, kaip tonas, trukmė, skiemens pozicija ir gretimi garsai. Veikiant sintezatoriui, norimas žodis sugeneruojamas iš tinkamiausių duomenų bazėje esančių kandidatų grandinės. Taip pasiekiamas didelis sintezuojamo balso natūralumas, nes įrašo nereikia papildomai apdoroti.

Pereinant iš antros kartos sintezės į trečios kartos sintezę buvo išplėstas difoną aprašančių požymių skaičius. Antros kartos sintezėje difonas aprašomas, kaip perėjimas iš vienos fonemos į kitą fonemą kartu su pagrindinio tono ir trukmių požymiais:

$$s_t = \begin{bmatrix} \text{STATE 1} & \begin{bmatrix} \text{PHONEME} & n \\ \text{F0} & 121 \\ \text{DURATION} & 50 \end{bmatrix} \\ \text{STATE 2} & \begin{bmatrix} \text{PHONEME} & t \\ \text{F0} & 123 \\ \text{DURATION} & 70 \end{bmatrix} \end{bmatrix}$$

Trečios kartos sintezėje difonas gali būti aprašomas kaip perėjimas iš vienos fonemos į kitą fonemą kartu su pagrindinio tono ir trukmių požymiais bei papildomais lingvistiniais požymiais, pvz., kirčiuotas-nekirčiuotas, esantis frazės pabaigoje-nesantis frazės pabaigoje:

$$s_t = \begin{bmatrix} \text{STATE 1} & \begin{bmatrix} \text{PHONEME} & n \\ \text{F0} & 121 \\ \text{DURATION} & 50 \\ \text{STRESS} & true \\ \text{PHARSE FINAL} & false \end{bmatrix} \\ \text{STATE 2} & \begin{bmatrix} \text{PHONEME} & t \\ \text{F0} & 123 \\ \text{DURATION} & 70 \\ \text{STRESS} & true \\ \text{PHARSE FINAL} & false \end{bmatrix} \end{bmatrix}$$

Akivaizdu, kad į difonų žodyną reiks įtraukti keletą to paties difono realizacijų, besiskiriančių minėtais papildomais požymiais.

USS sintezėje stengiamasi kuo mažiau modifikuoti vienetų bazę, t.y., laikomasi mažiausio vienetų modifikavimo principo (*principle of least modification*). 1996 m. A. Hunt ir A. Black pasiūlė algoritmą, kuris paplito trečios kartos sintezatoriuose. Skaitykime, kad naudojame difonų bazę. Specifikacija vadinsime difonų sąrašą S, o duomenų baze – difonų rinkinį U. USS algoritmo uždavinys – surasti difonų seką iš duomenų bazės U, kuri geriausiai atitinka specifikaciją S. Paieškai bus naudojamas tikslo įvertinimas (*target cost*) – atstumas tarp specifikacijos ir vieneto difonų bazėje. Antras matas yra sujungimo įvertinimas (*join cost*), skaičiuojamas tarp dviejų vienetų difonų bazėje pagal jų specifikacijas. Jungtinis viso sintezuojamo sakinio įvertinimas: (1.1)

$$C(U,S) = \sum_{t=1}^{T} T(u_t, s_t) + \sum_{t=1}^{T-1} J(u_t, u_{t+1}) \quad (1.1.)$$

Algoritmo uždavinys – surasti vienetų seką, minimizuojančią jungtinį įvertinimą (kainą):

$$U = argmin \left\{ \sum_{t=1}^{T} T(u_t, s_t) + \sum_{t=1}^{T-1} J(u_t, u_{t+1}) \right\} \quad (1.2)$$

Dažnai vietoje įvertinimo arba kainos naudojamas funkcijos terminas. Skaitykime, kad pasirinktoje sintezės sistemoje turime N fonų (fonemų) ir M skiemenų. USS naudojami tokie baziniai vienetų tipai:

1. Kadrai (*frames*) – atskiri kalbos fragmentai;
2. Būsenos (*states*) – fonų dalys;
3. Fonų pusės (*half-phones*) – tai pradinės arba galinės fonų pusės, gali būti 2N skirtingų fonų pusių;
4. Difonai (diphones) - perėjimai iš vienos fonemos į kitą fonemą, gali būti NxN difonų;
5. Fonai (*phones or phonemes*);
6. Demi-skiemenys (*demi-syllables*) – fonų pusių atitikmenys skiemenų lygyje, t.y., pradinės arba galinės skiemenų pusės, gali būti 2M skirtingų skiemenų pusių;
7. Di-skiemenys (*di-syllables*) - perėjimai iš vieno skiemens į kitą skiemenį, gali būti MxM di-skiemenų;
8. Skiemenys (syllables);
9. Žodžiai (words);
10. Frazės (phrases).

Europos kalbų sintezei plačiau naudojami fonai, difonai ir fonų pusės, tuo tarpu, pvz., kinų kalbos sintezei geriau tinka skiemenys, kadangi kalba turi skiemeninę struktūrą.

1.5.2. Balso dialogai

Balso dialogai – tai prietaisai, procesai, paslaugos, kurios yra pagrįstos įvairiomis kalbos apdorojimo technologijomis (kalbos generavimu, sinteze, suvokimu ir kt.), integruojant jas į kompiuterinės technologijos plotmę [16]. Kaip teigia John M. Smart savo nuolat atnaujinamame straipsnyje apie balso dialogus, nesvarbu kaip įvardinsime šias technologijas, jų reikšmės šiandienių prietaisų ir paslaugų kūrimui ir žmonijos kasdienybei bei pasiekimams nuginčyti negalime.

Balso dialogai ir iš jų besivystančios kitos technologijos yra mūsų visuomenės ateitis, bene didžiausias XXI amžiaus pirmos pusės atradimas ir arčiausiai tikslo esantis dirbtinio intelekto sistemų prototipas [17]. Balso dialogai tai programos, kurių pagalba žmonės gali "bendrauti" su kompiuteriais (ir ne tik) balsu, tai galimybė naudotis šių įrenginių paslaugomis, kai juos valdyti, tarkim, klaviatūra nėra labai patogu. Valdyti įrenginius balsu jūs galite bet kuriuo metu, bet kurioje vietoje [16]. Maksimalus bet kokio tipo balso paslaugos įgyvendinimas, galėtų būti paremtas dialogine struktūra, tačiau tam, kad sukurti optimaliai veikiančią sistemą reikia apjungti kalbos atpažinimo, sintezės, programavimo ir kitas žinias. Kaip teigia savo straipsnyje Zue ir Glass, kuriant tokio tipo sistemas susiduriama su beagle iššūkių: kaip tokia sistema turėtų veikti, kokį rezultatą turėtų gražinti? Kokios išimtys turėtų būti sistemoje? Šiems ir dar aibei klausimų turi būti surasti atsakymai, kad būtų galima sukurti interaktyvias dialogines sistemas [16]. Aišku tėra viena,

visapusiškas balso technologijų įgyvendinimas gali lemti didelius pasikeitimus įvairiose technologinėse srityse. Kaip teigia R. Maskeliūnas (Maskeliūnas,2006:4), *"Šiuo metų pasaulyje vis labiau plinta kalbine sąsaja išplėstos programos, ir tik laiko klausimas, kada galėsime pamiršti apie painias mygtukų kombinacijas ir diktuoti komandas balsu"*. Daugelyje sričių, balso dialogų tarp vartotojo ir sistemos įgyvendinimas, sumažina ir palengvina naudojimąsi sistemomis ir neįgaliems žmonėms: *„Daugeliui neįgaliųjų kalba - tai galbūt vienintelis bendravimo šaltinis. Balso dialogo galimybėmis išplėsti interneto tinklalapiai ir programos leidžia šiems žmonėms naršyti internete pateikiamą informaciją bei valdyti tam pritaikytus prietaisus tariant komandas balsu. Nereikia jokios brangios ir sudėtingos programinės įrangos."*(Maskeliūnas, 2006:5).Kalbiniai dialogai skirstomi į dialogus su *sistemos iniciatyva (systeminiciative)* ir su *mišria iniciatyva (mixediniciative)*. Dialoguose su sistemos iniciatyva vartotojas pateikia tik reikalaujamą informaciją, tuo tarpu dialoguose su mišria iniciatyva vartotojas gali pateikti ir papildomą informaciją, kuri bus reikalinga tolimesniuose dialogo etapuose

2. VAISTŲ KOMANDŲ PAVADINIMŲ ATPAŽINIMO EKSPERIMENTINIAI TYRIMAI

2.1. Apie projektą „Infobalsas“

Projekto tikslas yra sukurti hibridinę šnekos atpažinimo technologiją ir ją panaudoti kuriant pirmos informacinės paslaugos, naudojančios lietuvių šnekamosios kalbos komandų atpažinimą kaip pagrindinį žmogaus – kompiuterio sąsajos modalumą, prototipą. Ši balso komandų atpažinimo sistema yra skirta tam tikrai tikslinei grupei, gydytojams/farmacininkams, siekiant greitesnių rezultatų, ieškant informacijos duomenų bazėse, taip pat neprarandant kontakto su pacientu/klientu, nukreipiant dėmesį į kompiuterinės informacinės sistemos sąsają. Sukurta informacinė sistema suteiks galimybę gydytojui/farmacininkui balsu ištarti pageidaujamo vaisto pavadinimą arba gydytojų dažnai naudojamas frazes, susijusias su jų profesija, ir jos bus automatiškai atpažintos. Numatoma realizuoti ne mažiau kaip 1000 tokių balso komandų. Jos bus panaudotos paieškai farmacijos duomenų bazėse.

Siekiama, kad balso komandų atpažinimo tikslumas būtų ne mažesnis negu 90%. Jų atpažinimui bus naudojamas kitakalbis (pvz. anglų arba ispanų) kalbai sukurtas ir apmokytas balso atpažinimo variklis, adaptuojant jį lietuviškų balso komandų atpažinimui. Kadangi ne visos reikalingos balso komandos gali būti atpažintos pakankamai tiksliai, joms bus parengti alternatyvūs akustiniai ir atpažinimo modeliai, leidžiantys optimizuoti bendrą sistemos veikimo tikslumą.

Sukurta balso atpažinimo sistema bus atvira, t.y. potencialiai galės būti panaudojama ir pritaikoma kuriant kitas balso komandas naudojančioms paslaugoms, atitinkamai sistemą adaptavus ir apmokius kitam žodynui. Tikimasi, kad sukurtą balsu valdomą sistemą bus galima adaptuoti ir kitakalbiams vartotojams, kadangi naudojamas medicininių terminų ir vaistų garsynas yra sudarytas iš pavadinimų panašių į kitose kalbose vartojamus.

2.2. Projekto „Infobalsas“ garsynas

Siekiant įgyvendinti vykdomą projektą pirmiausia reikia surinkti daugiadiktorinį vaistų pavadinimų ir medicinos darbuotojų naudojamų komandų garsyną. Jo sudaryme dalyvavo 12 diktorių. Balso komandų įrašinėjimas vyko naudojant programą, veikiančia MS DOS operacinės sistemos aplinkoje. Visas garsynas yra suskirstytas į atskirus katalogus, po apytikriai 50 pavadinimų. Kiekviename kataloge yra:

audio failo turinį aprašantis pagalbinis failas „mano.txt“, kuriame nurodyta kiek kartų ištartas koks tekstas;

audio failas „mano.voc“ (16000 Hz, 16 bitų mono audio formatas) ir tas pats failas WAV formate „mano.wav“;

anotacijų failas „mano.zgl“, kuriame pateikti kiekvieno įrašo pradžios ir pabaigos adresai;

pagalbiniai failai, naudoti garsyno rinkimo ir tikrinimo metu.

Kiekvieno diktoriaus katalogo pavadinimas yra sudarytas iš diktoriaus lyties pirmosios raidės (M – vyras (male), F – moteris (female)) bei vardo ir pavardės pirmųjų 3 raidžių, pvz., FDANBRUZ, MKASRAT. Vaistų pavadinimų katalogai sudaryti iš pavadinimų pirmųjų raidžių, neviršijant 8 simbolių, pvz., kataloge „VAISTAI“ esantis pavadinimas „MIKARDIS“ yra kataloge pavadinimu „MIKARDIS“, o ilgesnis pavadinimas, pvz., „VALOKORDIN_LAŠAI“ yra kataloge, trumpesniu pavadinimu „VALOKORDIN“. Visi, garsyną sudarantys pavadinimai, pateikti prieduose:

2.3. Garsyno sudarymas

Siekiant įgyvendinti vykdomą projektą pirmiausia reikia surinkti daugiadiktorinį vaistų pavadinimų ir medicinos darbuotojų naudojamų komandų garsyną. Jo sudaryme dalyvavo 12 diktorių. Balso komandų įrašinėjimas vyko naudojant programą, veikiančia MS DOS operacinės sistemos aplinkoje. Visas garsynas yra suskirstytas į atskirus katalogus, po apytikriai 50 pavadinimų. Kiekviename kataloge yra: audio failo turinį aprašantis pagalbinis failas „mano.txt“, kuriame nurodyta kiek kartų ištartas koks tekstas;udio failas „mano.voc“ (16000 Hz, 16 bitų mono audio formatas) ir tas pats failas WAV formate „mano.wav“;anotacijų failas „mano.zgl“, kuriame pateikti

kiekvieno įrašo pradžios ir pabaigos adresai; pagalbiniai failai, naudoti garsyno rinkimo ir tikrinimo metu.

Kiekvieno diktoriaus katalogo pavadinimas yra sudarytas iš diktoriaus lyties pirmosios raidės (M – vyras (male), F – moteris (female)) bei vardo ir pavardės pirmųjų 3 raidžių, pvz., FDANBRUZ, MKASRAT. Vaistų pavadinimų katalogai sudaryti iš pavadinimų pirmųjų raidžių, neviršijant 8 simbolių, pvz., kataloge „VAISTAI“ esantis pavadinimas „MIKARDIS“ yra kataloge pavadinimu „MIKARDIS“, o ilgesnis pavadinimas, pvz., „VALOKORDIN_LAŠAI“ yra kataloge, trumpesniu pavadinimu „VALOKORDIN“. Visi, garsyną sudarantys pavadinimai, pateikti prieduose:

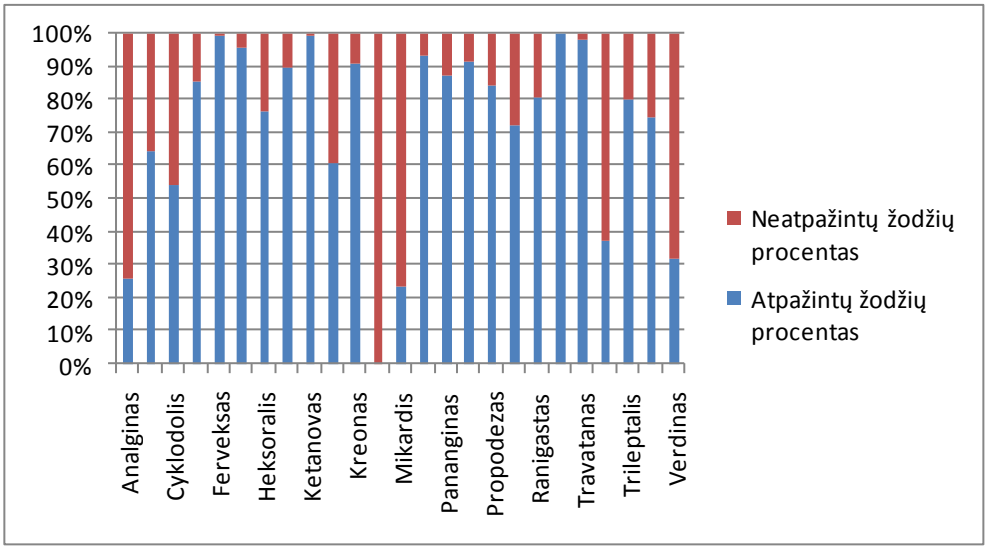
Vaistų pavadinimų garsynas yra kataloguose VAISTAI, VAISTAI2, VAISTAI3, VAISTAI4, VAISTAI5, VAISTAI6. Garsynų struktūrą galima matyti 4 priede.

2.4. Atpažinimo tyrimas su ispanų kalbos atpažintuvu

2.1. lentelė. Atpažintų žodžių procentas su ispanų kalbos atpažintuvu

Nr.	Žodis	Iš viso žodžių	Kiek atpažinta	Atpažinimo procentas
	Analginas	120	31	25,8%
	Bifovalis	120	77	64.2%
	Cyklodolis	120	65	54.2%
	Enarenalis	120	103	85,8%
	Ferveksas	120	119	99,1%
	Gastrovalis	120	115	95,8%
	Heksoralis	120	92	76,7%
	Hematogenas	120	108	90%
	Ketanovas	120	119	99,1%
	Ketonalis	120	73	60,8%
	Kreonas	120	109	90,8%
	Metforalis	120	0	0%
	Mikardis	120	28	23,3%
	Nebikardas	120	112	93,3%
	Pananginas	120	105	87,5%
	Preduktalis	120	110	91,7%
	Propodezas	120	101	84,2%
	Radireksas	120	87	72,5%
	Ranigastas	120	97	80,8%
	Trashisanas	120	120	100%
	Travatanas	120	118	98,3%
	Trentalis	120	45	37,5%
	Trileptalis	120	96	80%
	Valokordin_lašai	120	90	75%
	Verdinas	120	38	31,7%

Iš pateiktų duomenų matoma, kad bendras rezultatas su ispanišku kalbos atpažintuvu nepalyginamai geresnis už HTK, tačiau yra komandų kurios atpažintos beveik 100%. Pvz: vaistų balso komanda *trashisanas* iš 120 įrašų buvo atpažinti visi 120 įrašų. Žodžio *ketanovas* iš 120 įrašų buvo atpažinti 119 t.y 99,1%. *Travatanas* iš 120 buvo atpažinti 118 įrašų – t.y 98,8%.Iš viso 6 diktoriai 25 komandų sudaro 3000 ištarimų, iš kurių padaryta 842 klaidos. Bendras visų atpažintų įrašų procentas su ispanų kalbos atpažintuvu 72% (žr. 1.1. lent.)



2.1. pav. Atpažintų žodžių procentas su ispanų kalbos atpažintuvu

Tai gana aukšti rezultatai, leidžiantys daryti prielaidą, kad dar daugiau padirbėjus su modelių apmokymu, galima išgauti dar kokybiškesnius rezultatus. Iš gautų duomenų matoma, kad tik 5 komandos atpažintos mažesniu nei 90% tikslumu, net 8 komandos – daugiau nei 95,9% tikslumu, o tai yra beveik1/3 dalis testuotų komandų, ir 12 komandų patenkančių į atpažinimo tikslumo intervalą nuo 90% iki 95% (žr 2.1. pav.).

2.4.1. Gramatikų sudarymas

Kiekvienai bet kokiai atpažinimo sistemai reikia paruošti taip vadinamą sistemos „gramatiką“. Gramatikos sudarymas yra labai svarbus etapas, kadangi tai yra specialus failas, kuriame nurodoma, kaip tam tikra komanda (žodis) turėtų būti ištarta - tai vadinamosios transkripcijos, ir ką reikėtų išvesti į ekraną įvykus atpažinimui. Pirminės gramatikos sudarymui transkripcijos rašytos iš galvos, nesiremiant jokiais pagalbiniais transkripcijų generavimo įrankiais, beveik visos komandos aprašytos taip, kaip tariamos įprastai, tik nenaudojami lietuviški simboliai. Tokiu būdu siekiama patikrinti, kiek ir kokiu būdu galima pagerinti atpažinimo kokybę. Visą pirminės gramatikos failo turinį galima matyti 5 priede. Žemiau pateikiama ištrauka iš pirminės gramatikos failo:

<item><?MS_Grammar_Editor GroupWrap?>
<item> analginas</item>
<tag>$._value = "473 + 0 analginas"</tag></item>
<item><?MS_Grammar_Editor GroupWrap?>
<item> bifovalis</item>
<tag>$._value = "531 + 0 bifovalis"</tag></item>
<item><?MS_Grammar_Editor GroupWrap?>
<item> tsiklodolis</item>
<tag>$._value = "585 + 0 cYklodolis"</tag></item>

Antroji gramatika sudaroma pildant ją naujomis transkripcijomis, pasinaudojant sintezatoriumi, laisvai prieinamu internete, kurio kūrėjai yra *Smart Link Corporation*. Sintezatorius veikia *text-to-speech* principu. Transkripcijos kuriamos nesiremiant jokiu tiksliu metodu, žodis bandomas užrašyti įvairiais būdais, perklausomas, kaip jį ištaria sintezatorius, tobulinamas iki panašiausio ištarimo į užrašymą, o tada panaudojamos gramatikoje kaip alternatyvos. Taip atrodo ištrauka iš patobulintos gramatikos:

<item><?MS_Grammar_Editor GroupWrap?>
<item> cyklodolis</item>
<tag>$._value = "585 + 0 cyklodolis"</tag></item>
<item><?MS_Grammar_Editor GroupWrap?>
<item> tsiklodolis</item>

<tag>$._value = "585 + 1 cYklodolis"</tag></item>
<item><?MS_Grammar_Editor GroupWrap?>
<item> tsiklodolis</item>
<tag>$._value = "585 + 2 cYklodolis"</tag></item>

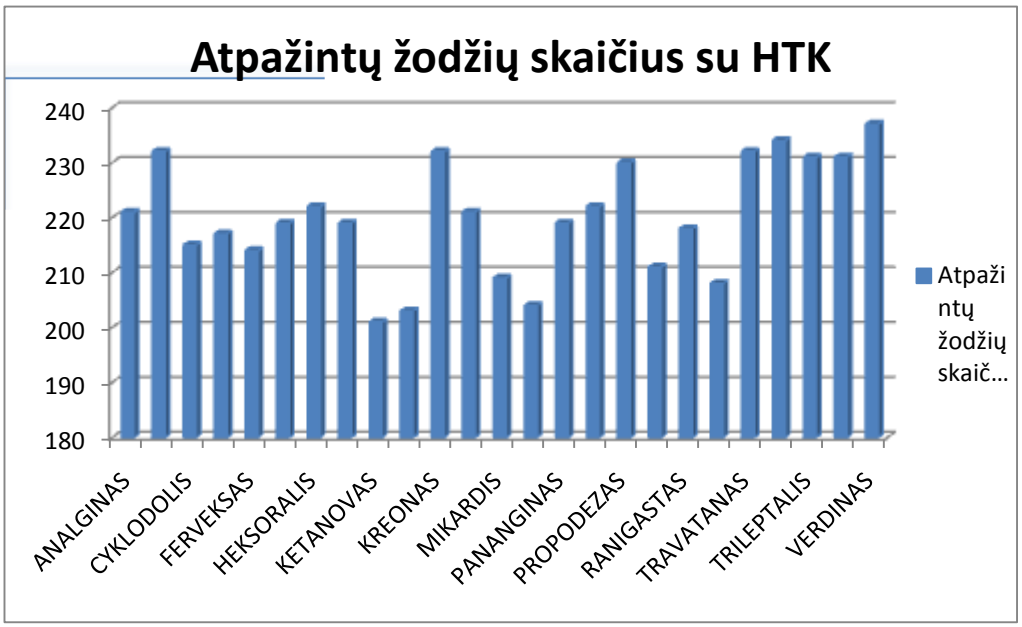
Trečioji gramatika sudarinėjama įtraukiant UPS transkripcijas.Jos generuojamos automatiniu būdu, panaudojant balso serverį ir jame įdiegtą *Microsoft Visual studio 2005* programą. Šiam darbui panaudojama antroji gramatika, su naujomis alternatyviomis transkripcijomis, kurios pakeičiamos UPS atributais. Sistema pateikia vieną arba daugiau galimų variantų, o vartotojas išsirenka norimas sugeneruotas transkripcijas ir jas įtraukia į gramatiką (šiame tyrime įtraukti visi sistemos siūlomi variantai).

Tyrimo metu bus naudojamos visos gramatikos ir stebimi testavimų pokyčiai, siekiant išgauti kuo aukštesnę atpažinimo kokybę.

2.5. Atpažinimo tyrimas su HTK paketu

Šiame skyriuje bus aptariamas HTK paketo įsisavinimas ir dalies garsyno testavimas remiantis šia technologija. Tyrimui naudojami dvylika diktorių: *fdanbru, fevemak, fginbar, fginpas, fieveig, mandmar, mkasrat, mromrut, msteant, mtomras, mtomstr, mvytru.* Pirmiesiems bandymams ir HTK paketo veikimo įsisavinimui, modelių apmokymas vyksta pagal diktoriaus *mvytrud*.

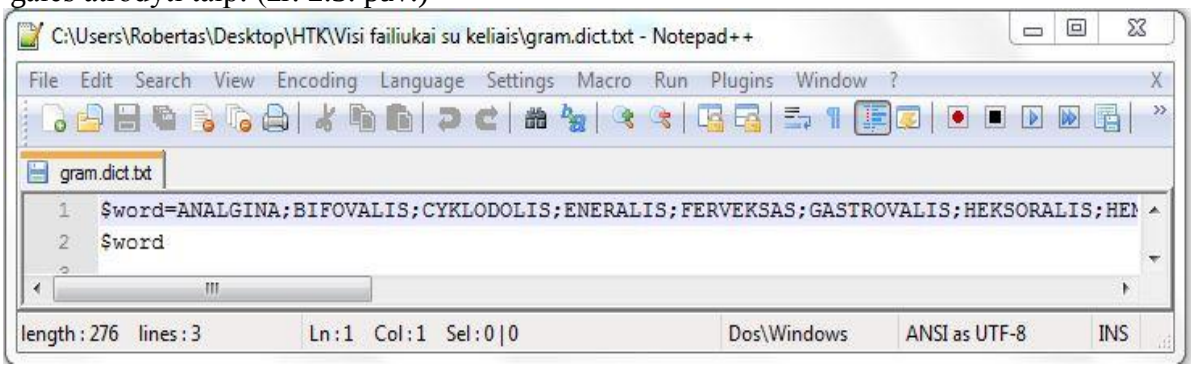
Naudojantis HTK atpažinimo įrankiu, buvo pasirinktos 25 - ios vaistų balso komandos - iš jau anksčiau atrinktų blogiausių komandų (kurios buvo sunkiai atpažįstamos kitos balso atpažinimo sistemos – ispaniško balso sintezatoriaus).Visos balso komandos buvo įrašytos 12 – os skirtingos lyties kalbėtojų po 20 kartų. Visoms balso komandos (vaistų pavadinimai) buvo paruošti gramatikos failai pagal HTK, taip pat suskaičiuoti požymiai ir apmokyti žodžių modeliai ir galiausiai atliekamas testavimas. Apmokius modelius ir atlikus testavimą gauti rezultatai, kurie parodo kiek procentaliai HTK sistema sugebėjo buvo atpažinti žodžių. Žemiau pateikiama statistika ir diagramos. Būtina paminėti, kad visi kalbėtojai šias komandas įrašinėjo studijoje, todėl triukšmas ir aplinkiniai trikdžiai buvo sumažinti iki minimumo. Taip pat pateikiami rezultatai atlikti su ispanišku balso sintezatoriumi (žr. 2.1. pav.).



2.1. pav. Atpažintų žodžių skaičius su HTK

Dirbant su HTK atpažinimo įrankiais, visų pirmiausia atpažinimo sistemai reikia paruošti taip vadinamą sistemos „gramatiką“: specialų failą, kuriame bus išvardinta kokie žodžiai ir kokia tvarka

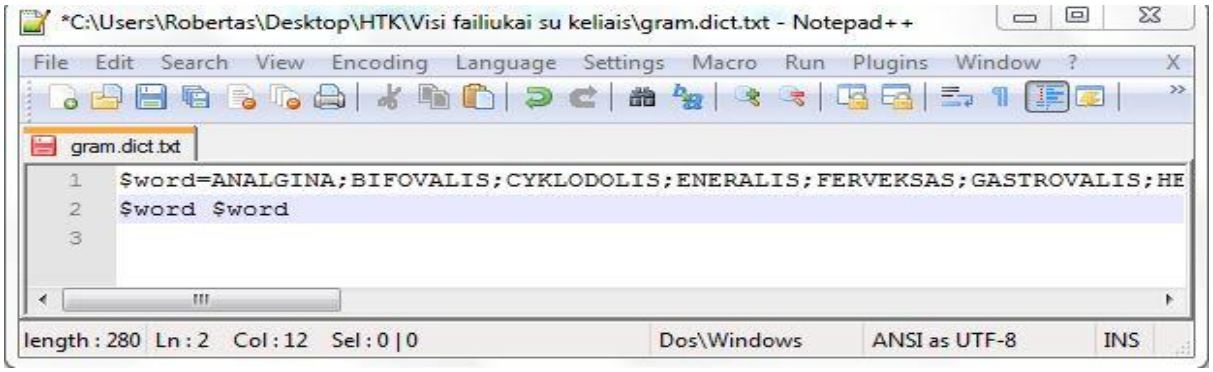
gali būti pateikti atpažinimo sistemai. Taigi vartotojas pirma turi sukurti failą, pavyzdžiui **gram.dict.** Tarkime mūsų sistema galės atpažinti 25 žodžius, kaip tai buvo daroma šiame darbe – analgina|bifovali|cyklodol|enarenal|ferveksa|gastrova|heksoral|hematoge|ketanov|ketonali|kreonas|metforal|mikardis|nebikard|panangin|preduct|propodez|radireks|ranigast|trachisa|travatan|trentali|trilepta|valokord|verdinas. Vieno ištarimo metu pasakomas tiktai vienas žodis. Tuomet failo gram.dict turinys galės atrodyti taip: (žr. 2.3. pav.)



2.2. pav. Gramdict turinys

Kitaip sakant, sistemai nurodome, kad pasakysime kažkokį žodį **word**, o tas žodis bus arba analginas bifovalis, cyklodolis, enarenalis, ferveksas, gastrovalis, heksoralis, hematogenas ir t.t. Jeigu to paties ištarimo metu sakytumėm du žodžius (pvz., analginasbifovalis), tai failo **gram.dict** turinys galėtų atrodyti taip: (žr. 2.4. pav.)

$word= analgina|bifovali|cyklodol|enarenal|ferveksa|
$word $word



2.3 pav. Gramdict turinys su dviem paeiliui žodžiais

Šiuo atveju žodžius reikia ištarti nepridedant jokių jungtukų, o tiesiog paeiliui tariant žodžius: analginas bifovalis. Tokiu atveju failas gram.dict padės atpažinimo sistemai atskirti du pavienius žodžius (žr. 2.3. pav.).

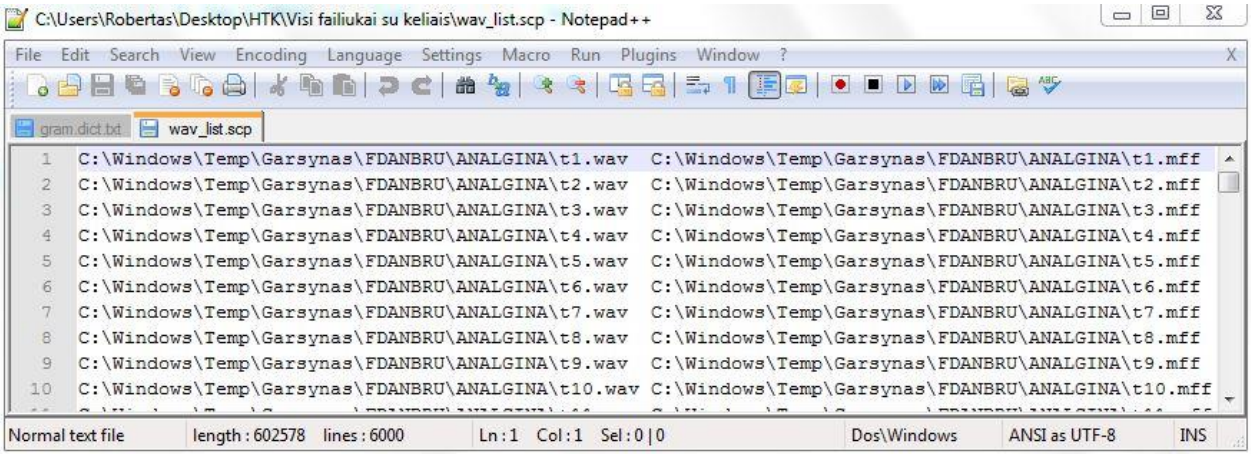
Jeigu vieno ištarimo metu sakome „analginas ir bifovalis“ ar pan., tuomet failo turinys turi atrodyti taip:

$word= analgina|bifovali|cyklodol|enarenal|ferveksa|
$word1= ir
$word$word1 $word

Šiuo atveju failas gram.dict padės atpažinti du skirtingus žodžius, kaip atskirus vienetus. Pasidarę reikalingą failą gram.dict susikuriame žodžių tinklo failą wordnet.txt, panaudodami programą HParse.exe: **HParsegram.dict wordnet.txt**

2.5.1. Požymių skaičiavimas

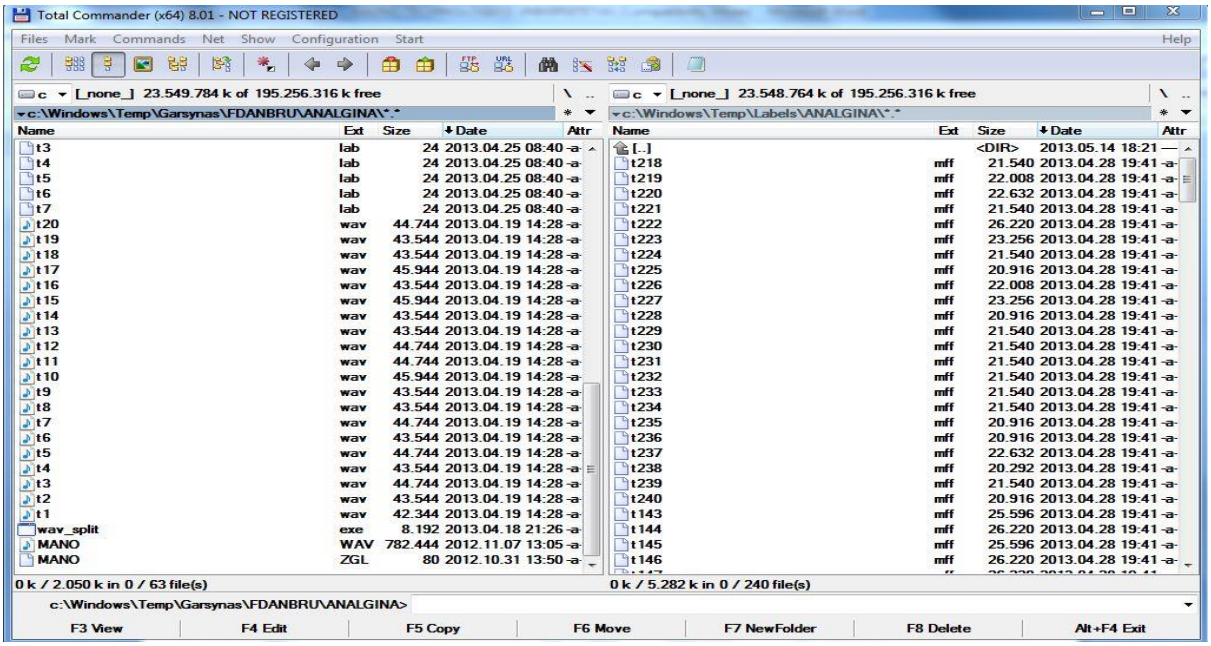
Šioje atpažinimo sistemoje yra naudojami ne patys įrašai, o iš tų įrašų apskaičiuoti požymiai. Kad būtų galima apskaičiuoti požymius reikia pasidaryti script-failus, kuriuose kompiuteriui reikia nurodyti kur yra įrašų failai. Script failuose išvardinami įrašų failai (ir kur jie randasi) ir požymių failai. Šiuo atveju buvo sukurtas failas wav_list.txt failas, kuriame buvo nurodyti visi esančių įrašų keliai, taip pat šalia tų kelių buvo nurodyta tas pats kelias į failą, tik su plėtiniu .mff vietoj .wav. (žr. 2.4. pav.)



2.3. pav. Failų konvertavimas

Konvertavus failus su plėtiniu .wav gauname failus su .mff. Konvertavimas vyksta, nurodžius komandą į Hcopy.exe failą ir paleidus jį su FAR arba totalcommander programomis. Komanda atrodo taip: HCopy –C config –S wav_list.scp c:\windowa\Temp\Garsynas\FDANBRU

ANALGINA.FaileConfigyrapateiktainformacijaapiepožymiųskaičiavimobūdus.Failewav_list.scpnurodomikeliai (žr. 2.5. pav.)



2.4. pav. Failų konvertavimas į .mff

2.5.2. Modelių failų parengimas

Modelių failų parengimas. Pirmas darbą rengiant atpažinimo sistemą darbui yra jos apmokymas. Pasirinktiems 25 vaistų pavadinimus, šiuo atveju „analginas, bifovalis, cyklodolis ir t.t.

buvo sukurti modelių failai kiekvienam žodžiui atskirai. Tada norint apmokyti pirmiausia reikia paruošti modelių failus pagal specialią formą. Taigi, kiekvienam žodžiui buvo sukurtas atskiras modelio failas. Modelių faile buvo nurodyta kiek būsenų bus naudojama žodžiui modeliuoti, kiek požymių bus naudojama požymių vektoriuje, nurodoma pradiniai vidurkių ir dispersijų vektoriai bei perėjimų tikimybių reikšmės. Matricos eilučių skaičius turi atitikti žodžio garsų skaičių su papildomomis dviejomis eilutėmis.

Būsenų skaičių būtina parinkti lygų garsų skaičiui žodyje, t.y. pvz. žodis „kreonas“ modeliuojamas 7 būsenomis, o žodis „verdinas“ – 8, kadangi pirma ir paskutinė būsenos neskaičiuojamos, tai modelių faile būsenų skaičius nurodomas dviem didesnis negu yra „realių“ būsenų, žodžiui „kreonas“ – 9, o žodžiui „verdinas“ - 10). Pateiktas pavyzdys: 2.6. pav.

```
~o
<STREAMINFO> 1 39
<VECSIZE> 39<NULLD><MFCC_D_A_E>
~h "hmm_kreonas"
<BEGINHMM>
<NUMSTATES> 9
<STATE> 2
<MEAN> 39
 -1.170338e+001 -1.634770e+000 -2.4562
<VARIANCE> 39
 4.126984e+001 3.707038e+001 2.793176e
```

2.5. pav. Būsenų skaičius

Modelių failas, kad žinotume, į kur nurodyti kelią, pavadiname tų žodžių vardais prieš tai pridėdami „hmm“, šiuo atveju „hmm_analginas“, t.y. žodžiui „analginas“ modelių failas vadinasi „hmm_analgnas“, o „cyklodolis“ – „hmm_cyklodol“. Atsidarius modelių failą 4 eilutėje nurodomas tmodelio pavadinimą, kuris turi būti lygiai toks kaip failo pavadinimas t.y. „hmm_analginas“ arba „hmm_cyklodol“ ir t.t

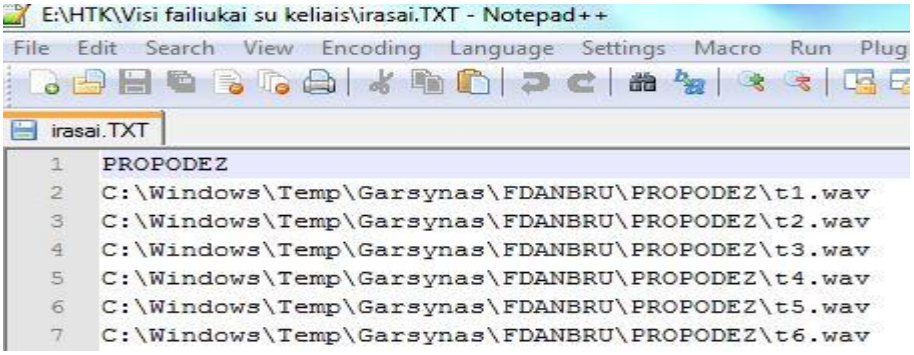
Pateikti modelių failų pavyzdžiai 5 ir 6 būsenoms, jei būsenų skaičius yra kitoks, reikia atitinkamai modifikuoti modelių failus, nurodant būsenų skaičių <NUMSTATES> ir pridedant/pašalinant atitinkamas būsenas ir tikimybių matricos eilutes/stulpelius: (žr. 2.7. pav.)

```
49 <TRANSP> 9
50  0.000000e+000 1.000000e+000 0.000000e+000 0.000000e+000 0.000000e+000 0.000000e+000 0.000000e+000 0.000000e+000 0.000000e+000
51  0.000000e+000 9.764478e-001 2.355223e-002 0.000000e+000 0.000000e+000 0.000000e+000 0.000000e+000 0.000000e+000 0.000000e+000
52  0.000000e+000 0.000000e+000 9.645538e-001 3.544620e-002 0.000000e+000 0.000000e+000 0.000000e+000 0.000000e+000 0.000000e+000
53  0.000000e+000 0.000000e+000 0.000000e+000 9.100529e-001 8.994710e-002 0.000000e+000 0.000000e+000 0.000000e+000 0.000000e+000
54  0.000000e+000 0.000000e+000 0.000000e+000 0.000000e+000 9.413793e-001 5.862069e-002 0.000000e+000 0.000000e+000 0.000000e+000
55  0.000000e+000 0.000000e+000 0.000000e+000 0.000000e+000 0.000000e+000 9.286913e-001 7.130872e-002 0.000000e+000 0.000000e+000
56  0.000000e+000 0.000000e+000 0.000000e+000 0.000000e+000 0.000000e+000 0.000000e+000 9.572650e-001 4.273505e-002 0.000000e+000
57  0.000000e+000 0.000000e+000 0.000000e+000 0.000000e+000 0.000000e+000 0.000000e+000 0.000000e+000 9.430145e+000 5.569855e+000
58  0.000000e+000 0.000000e+000 0.000000e+000 0.000000e+000 0.000000e+000 0.000000e+000 0.000000e+000 0.000000e+000 0.000000e+000
59 <ENDHMM>
```

2.6. pav. Matricos

2.5.3. Failų generavimas

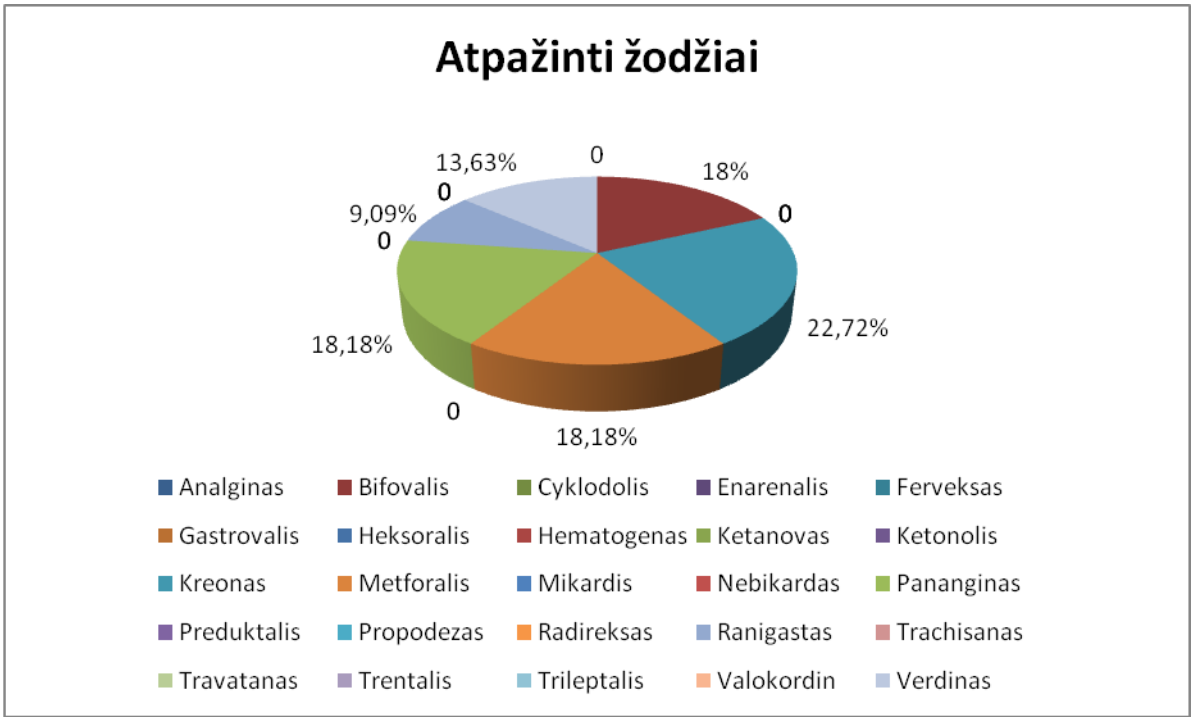
Generuojant failus HTK sistema reikalauja, kad kiekvienam garso įrašui turi būti sukurtas žymių failas, kaip anksčiau buvo minėta. Jų sukūrimui buvo naudojama programėle zymiu_generavimas.exe. Programėlę paleidus šalia sukuriamas failas irasai.txt, kurio pirmoje eilutėje užrašytas žodžio pavadinimas, o sekančiose eilutėse išvardinti duoto žodžio įrašai (po paskutinio įrašo įterpta viena tuščia eilutė). Tokiu būdu paleidus šią programėlę naudojant komandą *„Zymiu_generavimas.exe –T 1 -S train_props.txt -l* PROPODEZ *-L c:\windows\Temp\Garsynas\FDANBRU\PROPODEZ_hmm_propodezas“*gauname reikiamus žymių failus c:\Windows\Temp\Garsynas\FDANBRU\PROPODEZ\t1.lab Pvz. žodžiui PROPODEZ failo irasai.txt turinys atrodo taip: žr 2 pav.



2.7. pav. įrašai.txt turinys

2.5.4. Atpažinimo rezultatai

Naujai įrašyti 125 žodžiai, atliktas modelio apmokymas, atlikus testavimą, atpažinimo rezultatai 18,4%. Diagramoje pateikiama, kokių žodžių HTK sistema procentaliai daugiausiai atpažįsta. Tendencija yra tokia, kad esant daugiau kalbėtojų t.y. įrašius daugiau žodžių HTK programos atpažinimo rezultatai gerėja, ką ir galime pastebėti žemiau pavaizduotoje diagramoje (žr. 2.9. pav.)



2.8. pav. Atpažinti po apmokymo

Ši diagrama parodo rezultatus įrašius apmokytus modelius. Visi 25 medicininiai terminai buvo įrašyti po 5- is kartus paprastu mikrofonu – iš viso 125 garsiniai įrašai iš kurių buvo atpažinta 23 žodžiai. Viršuje pavaizduota kiek kokių žodžių buvo atpažinta (procentais). **Žr 2 pav.** Įsigilinus į diagramą galima teigti, jog panašiai skambantys žodžiai, turintys tą pačia šaknį ar galūnę dažniausiai sumaišomi ir atpažįstami kaip vienas ir tas pats žodis. Tarkim diagramoje matome, kad žodis kreonas buvo atpažintas geriausiai, kadangi analogiškų – panašių žodžių į jį nėra, todėl HTK sistemai jį atpažinti buvo lengviausiai. Tarkim žodžių ketanovas ir ketanolis sistemai nepavyko išvis atpažinti, todėl analogiškai buvo atliktas antrasis bandymas esant praktiškai idealioms sąlygoms.

3. REZULTATŲ APIBENDRINIMAS IR IŠVADOS

1. Atlikus teorinių šaltinių analizę, galima teigti, kad ne tik daugybė mokslininkų, bet ir įvairių paslaugų, pramonės sričių specialistų, yra vis labiau suinteresuoti kalbos atpažinimo tyrimais ir pritaikymu praktikoje. Tokia galimybė gali būti ne tik pramoginio pobūdžio, bet ir pagalbinė priemonė medicinoje.
2. Išanalizavus HTK balso atpažinimo programos veikimo principą, paaiškėjo, jog norint, kad sistema atpažintų tam tikrą žodį, reikia parengti gramatikos failą, kuriame turime nurodyti žodžių bazę, taip pat parengti *script* failus, kuriuose reikia nurodyti kelius, kur patalpinti įrašai, taip pat parengti modelių failus.
3. Kadangi HTK sistemoje naudojami ne patys įrašai, o iš tų įrašų apskaičiuoti požymiai buvo apskaičiuoti požymiai ir padaryti *script*-failai, kad sistema sugebėtų rasti kur yra įrašų failai. Tokiu būdu HTK sistema lengvai suranda esamus įrašus, apdoroja juos ir atpažįsta.
4. Įrašius žodžius kurie buvo apmokyti pagal kitus diktorius natūraliomis HTK sistema gan sunkiai atpažino žodžius. Bendras visų atpažintų žodžių procentas yra 18,4%.
5. Atsižvelgiant į skirtingas aplinkybes galima teigti, kad HTK programos patikimumas pakankamai pakankamai aukštas esant idealioms sąlygoms, tačiau dar daug tobulintinas esant natūralioms sąlygoms.
6. Naudojant ispanų kalbos atpažintuvas *Microsoft Speech Recognizer 8.0* bendras rezultatas nepalyginamai geresnis už HTK 72%. Tačiau yra komandų kurios atpažintos beveik 100 %. Iš viso 6 diktoriai 25 komandų sudaro 3000 ištarimų, iš kurių be prailgintų pauzių padaryta 842 klaidos.

4. LITERATŪROS SĄRAŠAS

[1] J. H. M. Daniel Jurafsky (2000). Speech and Language Processing, An introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition. Prentice Hall, Upper Saddle River, New Jersey 07458.

[2] Fosberg, M. (2003) Why is speech recognition difficult? Chalmers University,Department of Computer:Prieiga per internetą: http://www.speech.kth.se/~rolf/gsltpapers/MarkusForsberg.pdf

[3] Rasa Lileikytė (2012). ŠNEKOS ATPAŽINIMO POŽYMIŲ KOKYBĖS VERTINIMAS: daktaro disertacija: Technologijos mokslai, informatikos inžinerija (07T) / Vilniaus universitetas. Prieiga per internetą: http://www.mii.lt/files/mii_dis_2012_lileikyte.pdf

[4] Ringelienė Ž., Filipovič M. Žodžių atpažinimo, grįsto paslėptaisiais Markovo modeliais, vizualizavimo ir analizės programinė įranga. ISSN 1392-0561. Prieiga per internetą: http://www.leidykla.eu/fileadmin/informacijos_mokslai/2011-56/63-72.pdf

[5] Rudžionis, A. (2001) Balso technologijų taikymo lietuvių kalbai analizė ir perspektyvinių veiklos krypčių pagrindimas. Prieiga per internetą: http://www.likit.lt/frames/balso_tech/balsotech_st.htm

[6] Balvočius B., Telksnys L. (2003). Lietuvių kalbos kompiuteriniai tyrimai (IX sekcija). Prieiga per internetą: http://www.elibrary.lt/resursai/konferencijos/ktu_01/it_2003/sekcija09.pdf

[7] Kasparaitis, P. (2001). Lietuvių kalbos kompiuterinė sintezė: daktaro disertacija: fiziniai mokslai, informatika (09P) / Vilniaus universitetas.

[8] Kasparaitis P. (2005). Kompiuterinės lingvistikos paskaitų konspektai. Prieiga per internetą: http://www.mif.vu.lt/~pijus/cl/cl.htm

[9] Laurinčiukaitė, S. (2008) Lietuvių šnekos atpažinimo akustinis modeliavimas: daktaro disertacija: technologijos mokslai, informatikos inžinerija 07T / Vilniaus Gedimino technikos universitetas. Prieiga per internetą: www.mii.lt/files/mii_dis_08_laurinciukaite.pdf

[10] Rudžionis A., Rudžionis V., Žvinys P. (2000) Lietuvių šnekamosios kalbos garsynas LTDIGITS: rezultatai ir problemos. Informacinės technologijos 2000. ISBN 9986-13-826-4. Kaunas, Technologija, 2000, p.162 – 166.

[11] Pranevičius, H., Raudys, Š., Rudžionis, A., Rudžionis, V., Ratkevičius, K., Sakalauskaitė, J., Makackas, D. (2008). Agentinių sistemų modeliai: mokomoji knyga. Vilnius: Mokslo aidai. 225 p. ISBN 978-9955-591-55-9.

[12] Kazlauskienė A., Raškinis G., Vaičiūnas A. (2010).Automatinis lietuvių kalbos žodžių skiemenavimas, kirčiavimas, transkribavimas. Prieiga per internetą: Interaktyvus [žiūrėta 2013-03-01]. http://issuu.com/vduleidykla/docs/automatinistranskribavimas

[13] Maskeliūnas R. (2009). Lietuviškų balso komandų atpažinimas daugybinių transkripcijų pagrindu: daktaro disertacija: technologijos mokslai, informatikos inžinerija – 07T / Kauno technologijos universitetas. Kaunas. 159 p.

[14] Maskeliūnas, R., Rudžionis, A., Rudžionis, V., Ratkevičius, K. (2010). Modeling of call services for public sector // Electronics and electrical engineering. Kaunas: Technologija. No. 4(100), p. 81-86. ISBN 1392-1215

[15] Young, S. ir kt. The HTK Book (for HTK Version 3.4), 2006, 2 p.

[16] Zue V.W., Glass R.J, (2000) Conversational Interfaces: Advances and Challenges, PUBLISHED IN PROCEEDINGS OF THE IEEE, VOL. 88, NO. 8, 1166-1180, AUGUST, 2000.[žiūrėta 2013-03-23]Prieiga per internetą: http://people.csail.mit.edu/jrg/2000/proc01.pdf

[17] Smart, M.J.(2003 – 2010), The Conversational Interface: Our Next Great Leap Forward.: Prieiga per internetą: [žiūrėta 2013-03-05] http://www.accelerationwatch.com/lui.html

[18] Baranauskas,V. (2006). Artikuliacinės kalbos sintezavimas: magistro darbas, elektronika / Šiaulių universitetas, Šiauliai. Interaktyvus [žiūrėta 2013-03-09]. Prieiga per internetą: http://vddb.library.lt/fedora/get/LT-eLABa-0001:E.02~2006~D_20060612_004117-95470/DS.005.0.01.ETD

[19] Dobrovolskis, M., Zonys, K., (2005). Šnekos atpažinimas: magistro darbas elektronika / Šiaulių universitetas. Šiauliai. Interaktyvus [žiūrėta 2013-02-20]. Prieiga per internetą: http://vddb.library.lt/fedora/get/LT-eLABa-0001:E.02~2005~D_20050614_154005-58155/DS.005.0.01.ETD

[20] Vaičiūnas, A.; Kaminskas, V.; Raškinis G. Statistical Language Models of Lithuanian Based on Word Clustering and Morphological Decomposition. Informatica, 2004, vol. 15, no. 4, p.565-580.

5. PRIEDAI

5.1. Priedas. Rezultatai gauti su HTK sistema po apmokymo

Vaistai

Analginasatpažino, kaip 0 18400000 PANANGIN -11107.753906	Analginas2 atpažino, kaip 0 19000000 PANANGIN -11541.107422
Analginas3 atpažino, kaip 0 216A00000 PANANGIN -12109.920898	Analginas4 atpažino, kaip 0 18400000 PANANGIN -11064.254883
Analginas5 atpažino, kaip 0 45900000 PANANGIN -25554.400391	Bifovalis, atpažino, kaip 0 51000000 PANANGIN -28553.169922
Bifovalis2, atpažino, kaip 0 20900000 BIFOVALI -12207.494141	Bifovalis3, atpažino, kaip 0 17100000 BIFOVALI -10391.355469
Bifovalis4, atpažino, kaip 0 23500000 BIFOVALI -13712.245117	Bifovalis5, atpažino, kaip 0 21600000 BIFOVALI -12654.519531
Cyklodolis, atpažino, kaip 0 20900000 KREONAS -12232.073242	Cyklodolis2, atpažino, kaip 0 20300000 PANANGIN -11742.466797
Cyklodolis3, atpažino, kaip 0 24100000 PANANGIN -14040.577148	Cyklodolis4, atpažino, kaip 0 20900000 KREONAS -12001.456055
Cyklodolis5, atpažino, kaip 0 20900000 KREONAS -12124.489258	Enarenalis, atpažino, kaip 0 17700000 PANANGIN -10593.399414
Enarenalis2, atpažino, kaip 0 24100000 PANANGIN -14038.321289	Enarenalis3, atpažino, kaip 0 22200000 KREONAS -13970.332031
Enarenalis4, atpažino, kaip 0 21600000 PANANGIN -12750.792969	Enarenalis5, atpažino, kaip 0 19600000 PANANGIN -11399.375000
Ferveksas, atpažino, kaip 0 16400000 KREONAS -9810.198242	Ferveksas2, atpažino, kaip 0 20300000 TRENTALI -11452.596680
Ferveksas3, atpažino, kaip 0 16400000 TRENTALI -9181.907227	Ferveksas4, atpažino, kaip 0 20300000 PANANGIN -11396.964844
Ferveksas5, atpažino, kaip 0 20300000 TRENTALI -11718.994141	Gastrovalis, atpažino, kaip 0 17700000 METFORAL -10427.062500
Gastrovalis2, atpažino, kaip 0 17100000 METFORAL -9962.907227	Gastrovalis3, atpažino, kaip 0 22200000 METFORAL -12825.079102
Gastrovalis4, atpažino, kaip 0 20900000 METFORAL -11999.547852	Gastrovalis5, atpažino, kaip 0 17700000 METFORAL -10668.581055
Heksoralisatpažino, kaip 0 20300000 METFORAL -12305.557617	Heksoralis2 atpažino, kaip 0 19000000 PANANGIN -10501.791016
Heksoralis3 atpažino, kaip 0 21600000 PANANGIN -12559.680664	Heksoralis4 atpažino, kaip 0 19600000 METFORAL -11209.496094
Heksoralis5 atpažino, kaip 0 20900000 TRENTALI -11953.057617	Hematogenasatpažino, kaip 0 22200000 PANANGIN -12969.899414
Hematogenas atpažino2, kaip 0 20900000 PANANGIN -12533.566406	Hematogenasatpažino, kaip 0 18400000 KREONAS -11367.322266
Hematogenasatpažino, kaip 0 22800000 PANANGIN -13186.293945	Hematogenasatpažino, kaip 0 22800000 PANANGIN -13713.228516
Ketanovasatpažino, kaip 0 17700000 KREONAS -10201.807617	Ketanovas2 atpažino, kaip 0 19000000 KREONAS -11676.104492
Ketanovas3 atpažino, kaip 0 19000000 KREONAS -10896.141602	Ketanovas4 atpažino, kaip 0 18400000 KREONAS -10603.183594

Ketanovas5 atpažino kaip 0 22200000 KREONAS -12891.868164	Ketonolis atpažino kaip 0 20900000 KREONAS -11910.850586
Ketonolis2 atpažino kaip 0 23500000 PANANGIN -13757.623047	Ketonolis3 atpažino kaip 0 22800000 KREONAS -13195.596680
Ketonolis4 atpažino kaip 0 23500000 KREONAS -13602.589844	Ketonolis5 atpažino kaip 0 20300000 KREONAS -12357.260742
Kreonas atpažino kaip 0 14500000 KREONAS -8294.620117	Kreonas2 atpažino kaip 0 15200000 KREONAS -8669.776367
Kreonas3 atpažino kaip 0 19600000 KREONAS -11249.813477	Kreonas4 atpažino kaip 0 16400000 KREONAS -9562.144531
Kreonas4 atpažino kaip 0 16400000 KREONAS -9559.176758	Metforalis atpažino kaip 0 17700000 METFORAL -10995.542969
Metforalis2 atpažino kaip 0 18400000 METFORAL -11591.939453	Metforalis3 atpažino kaip 0 18400000 METFORAL -11082.424805
Metforalis4 atpažino kaip 0 21600000 METFORAL -13518.058594	Metforalis5 atpažino kaip 0 20300000 MIKARDIS -12764.503906
Mikardis atpažino, kaip 0 19000000 PANANGIN -11318.145508	Mikardis2 atpažino, kaip 0 19000000 PANANGIN -11274.426758
Mikardis3 atpažino, kaip 0 19600000 PANANGIN -11562.102539	Mikardis4 atpažino, kaip 0 21600000 PANANGIN -12469.838867
Mikardis5 atpažino, kaip 0 20900000 PANANGIN -12516.704102	Nebikardas atpažino kaip 0 20300000 MIKARDIS -12303.245117
Nebikardas2 atpažino kaip 0 25400000 KREONAS -15586.078125	Nebikardas3 atpažino kaip 0 22200000 PANANGIN -13462.083984
Nebikardas4 atpažino kaip 0 20900000 PANANGIN -12395.264648	Nebikardas5 atpažino kaip 0 21600000 PANANGIN -13168.700195
Pananginasatpažino kaip 0 21600000 PANANGIN -12403.594727	Pananginas2 atpažino kaip 0 20300000 PANANGIN -12419.407227
Pananginas3 atpažino kaip 0 22800000 PANANGIN -12793.009766	Pananginas4 atpažino kaip 0 17700000 PANANGIN -10289.790039
Pananginas5 atpažino kaip 0 17100000 KREONAS -10851.739258	Preduktalisatpažino kaip 0 20900000 PANANGIN -11953.784180
Preduktalis2 atpažino kaip 0 17700000 MIKARDIS -10948.484375	Preduktalis3 atpažino kaip 0 17700000 METFORAL -10429.298828
Preduktalis4 atpažino kaip 0 17700000 MIKARDIS -11307.728516	Preduktalis5 atpažino kaip 0 19600000 METFORAL -11544.754883
Propodezas atpažino kaip 0 20900000 RANIGAST -12529.533203	Propodezas2 atpažino kaip 0 19600000 RANIGAST -12186.293945
Propodezas3 atpažino kaip 0 21600000 PANANGIN -12780.717773	Propodezas4 atpažino kaip 0 20300000 RANIGAST -12582.105469
Propodezas5 atpažino kaip 0 21600000 KREONAS -13091.962891	Radireksas atpažino kaip 0 22200000 RANIGAST -13355.509766
Radireksas2 atpažino kaip 0 20300000 KREONAS -12203.677734	Radireksas3 atpažino kaip 0 21600000 RANIGAST -12938.788086
Radireksas4 atpažino kaip 0 21600000 RANIGAST -13058.068359	Radireksas5 atpažino kaip 0 16400000 RANIGAST -10277.884766
Ranigastasatpažino kaip 0 20900000 KREONAS -12713.111328	Ranigastas2 atpažino kaip 0 19600000 KREONAS -11871.159180
Ranigastas3 atpažino kaip 0 17700000 RANIGAST -11208.419922	Ranigastas4 atpažino kaip 0 20900000 KREONAS -13415.860352
Ranigastas5 atpažino kaip 0	Trachisanasatpažino kaip 0

18400000 RANIGAST -10824.168945	19600000 PANANGIN -11970.075195
Trachisanas2 atpažino kaip 0 19600000 RANIGAST -12373.206055	Trachisanas3atpažino kaip 0 22800000 VERDINAS -13802.297852
Trachisanas4 atpažino kaip 0 19600000 PANANGIN -11970.075195	Trachisanas5 atpažino kaip 0 19600000 KREONAS -11952.736328
Travatanasatpažino kaip 0 19600000 KREONAS -12484.407227	Travatanas2 atpažino kaip 0 24100000 KREONAS -14188.668945
Travatanas3 atpažino kaip 0 18400000 PANANGIN -10929.377930	Travatanas4 atpažino kaip 0 19600000 KREONAS -12054.570313
Travatanas5 atpažino kaip 0 19000000 KREONAS -12133.433594	Trentalisatpažino kaip 0 20300000 PANANGIN -11980.399414
Trentalis2 atpažino kaip 0 20300000 PANANGIN -11736.964844	Trentalis3 atpažino kaip 0 22800000 METFORAL -13905.799805
Trentalis4 atpažino kaip 0 23500000 PANANGIN -13412.955078	Trentalis5 atpažino kaip 0 19000000 PANANGIN -10372.554688
Trileptalis atpažino kaip 0 20900000 TRENTALI -12173.528320	Trileptalis2 atpažino kaip 0 19000000 PANANGIN -11037.807617
Trileptalis3 atpažino kaip 0 23500000 PANANGIN -13625.422852	Trileptalis4 atpažino kaip 0 18400000 PANANGIN -10525.945313
Trileptalis5 atpažino kaip 0 19600000 PANANGIN -11372.466797	Valokordin atpažino kaip 0 18400000 PANANGIN -11079.150391
Valokordin2 atpažino kaip 0 16400000 KREONAS -10168.015625	Valokordin3 atpažino kaip 0 62500000 PANANGIN -35633.996094
Valokordin4 atpažino kaip 0 19600000 PANANGIN -11836.263672	Valokordin5 atpažino kaip 0 17700000 KREONAS -10887.952148
Verdinas atpažino kaip 0 13900000 VERDINAS -8428.829102	Verdinas2 atpažino kaip 0 19600000 PANANGIN -11348.719727
Verdinas3 atpažino kaip 0 20300000 VERDINAS -11881.480469	Verdinas4 atpažino kaip 0 20300000 VERDINAS -11296.574219
Verdinas5 atpažino kaip 0 25400000 PANANGIN -14334.386719	Iš 125 apmokytų modelių atpažinta 23

5.2. priedas. Atpažinimo gramatika

Kiekvienai bet kokiai atpažinimo sistemai reikia paruošti taip vadinamą sistemos „gramatiką“. Gramatikos sudarymas yra labai svarbus etapas, kadangi tai yra specialus failas, kuriame nurodoma, kaip tam tikra komanda (žodis) turėtų būti ištarta - tai vadinamosios transkripcijos, ir ką reikėtų išvesti į ekraną įvykus atpažinimui. Pirminės gramatikos sudarymui transkripcijos rašytos iš galvos, nesiremiant jokiais pagalbiniais transkripcijų generavimo įrankiais, beveik visos komandos aprašytos taip, kaip tariamos įprastai, tik nenaudojami lietuviški simboliai. Tokiu būdu siekiama patikrinti, kiek ir kokiu būdu galima pagerinti atpažinimo kokybę.

```
<grammar xmlns:sapi="http://schemas.microsoft.com/Speech/2002/06/SRGSExtensions"
 xml:lang="ES-ES" tag-format="semantics-ms/1.0" version="1.0" mode="voice"
 xmlns="http://www.w3.org/2001/06/grammar" sapi:alphabet="x-microsoft-ups">
<rule id="Rule" scope="public"><one-of>


<item><?MS_Grammar_Editor GroupWrap?>
<item> analginas</item>
```

<tag>$._value = "473 + 0 analginas"</tag></item>

<item><?MS_Grammar_Editor GroupWrap?>
<item> bifovalis</item>
<tag>$._value = "531 + 0 bifovalis"</tag></item>

<item><?MS_Grammar_Editor GroupWrap?>
<item> tsiklodolis</item>
<tag>$._value = "585 + 0 cYklodolis"</tag></item>

<item><?MS_Grammar_Editor GroupWrap?>
<item> enarenalis</item>
<tag>$._value = "585 + 0 enarenalis"</tag></item>

<item><?MS_Grammar_Editor GroupWrap?>
<item>ferveksas</item>
<tag>$._value = "647 + 0 ferveksas"</tag></item>

<item><?MS_Grammar_Editor GroupWrap?>
<item> gastrovalis</item>
<tag>$._value = "430 + 0 gastrovalis"</tag></item>

<item><?MS_Grammar_Editor GroupWrap?>
<item>heksoralis</item>
<tag>$._value = "699 + 0 heksoralis"</tag></item>

<item><?MS_Grammar_Editor GroupWrap?>
<item> hematogenas</item>
<tag>$._value = "435 + 0 hematogenas"</tag></item>

<item><?MS_Grammar_Editor GroupWrap?>
<item> ketanovas</item>
<tag>$._value = "444 + 0 ketanovas"</tag></item>

<item><?MS_Grammar_Editor GroupWrap?>
<item> ketonalis</item>
<tag>$._value = "601 + 0 ketonalis"</tag></item>

<item><?MS_Grammar_Editor GroupWrap?>
<item> kreonas</item>
<tag>$._value = "602 + 0 kreonas"</tag></item>

<item><?MS_Grammar_Editor GroupWrap?>
<item> metforalis</item>
<tag>$._value = "497 + 0 metforalis"</tag></item>

<item><?MS_Grammar_Editor GroupWrap?>
<item>mikardis</item>
<tag>$._value = "709 + 0 mikardis"</tag></item>

<item><?MS_Grammar_Editor GroupWrap?>
<item>nebikardas</item>

<tag>$._value = "661 + 0 nebikardas"</tag></item>

<item><?MS_Grammar_Editor GroupWrap?>
<item> pananginas</item>
<tag>$._value = "508 + 0 pananginas"</tag></item>

<item><?MS_Grammar_Editor GroupWrap?>
<item> preduktalis</item>
<tag>$._value = "458 + 0 preduktalis"</tag></item>

<item><?MS_Grammar_Editor GroupWrap?>
<item> propodezas</item>
<tag>$._value = "560 + 0 propodezas"</tag></item>

<item><?MS_Grammar_Editor GroupWrap?>
<item> radireksas</item>
<tag>$._value = "510 + 0 radireksas"</tag></item>

<item><?MS_Grammar_Editor GroupWrap?>
<item>ranigastas</item>
<tag>$._value = "668 + 0 ranigastas"</tag></item>

<item><?MS_Grammar_Editor GroupWrap?>
<item>trashisanas</item>
<tag>$._value = "723 + 0 trashisanas"</tag></item>

<item><?MS_Grammar_Editor GroupWrap?>
<item>travatanas</item>
<tag>$._value = "725 + 0 travatanas"</tag></item>

<item><?MS_Grammar_Editor GroupWrap?>
<item>trentalis</item>
<tag>$._value = "674 + 0 trentalis"</tag></item>

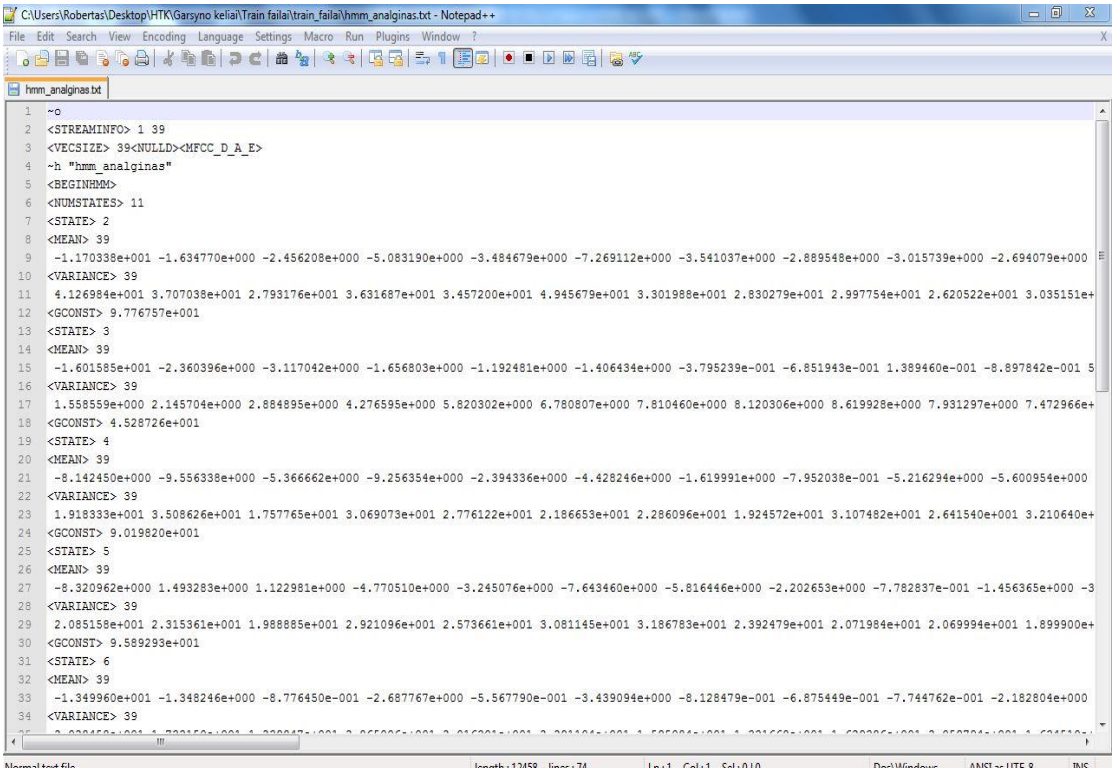
<item><?MS_Grammar_Editor GroupWrap?>
<item>trileptalis</item>
<tag>$._value = "726 + 0 trileptalis"</tag></item>

<item><?MS_Grammar_Editor GroupWrap?>
<item>valokordin_lashai</item>
<tag>$._value = "728 + 0 valokordin_lashai"</tag></item>

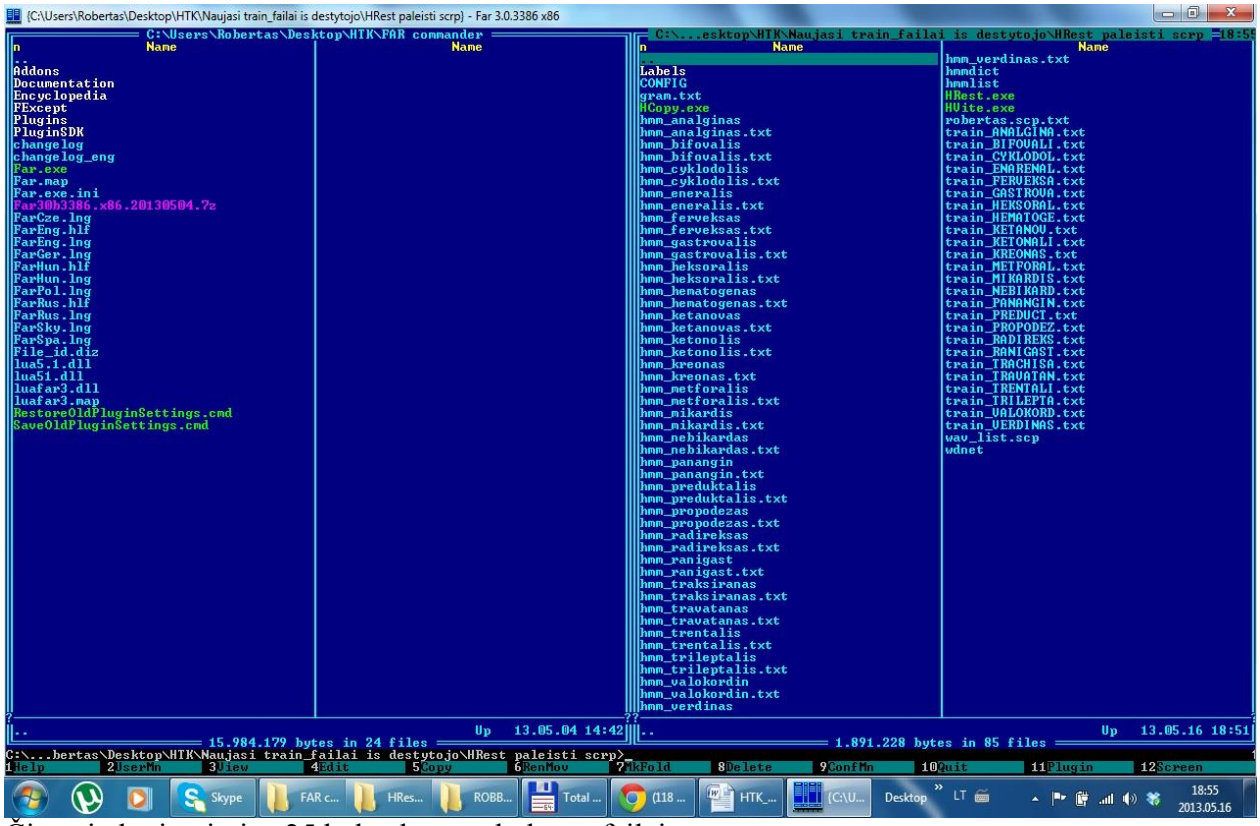
<item><?MS_Grammar_Editor GroupWrap?>
<item> verdinas</item>
<tag>$._value = "576 + 0 verdinas"</tag></item>

</one-of></rule></grammar>

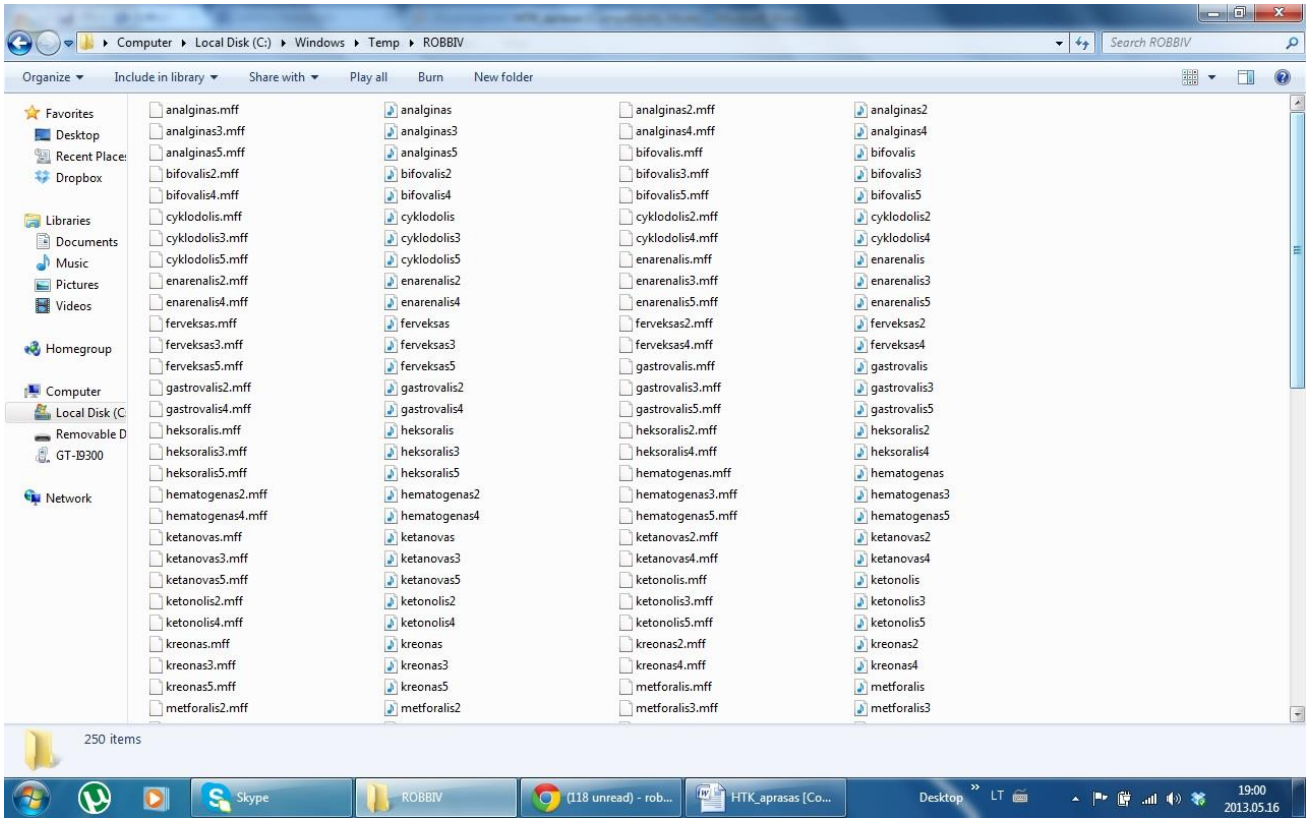
5.3. priedas. HTK sistemos paketas ir parengti failai



Kiekvienam žodžiui turi būti sukurtas atskiras modelio failas. Modelių faile reikia nurodyti kiek būsenų bus naudojama žodžiui modeliuoti, kiek požymių bus naudojama požymių vektoriuje, nurodoma pradiniai vidurkių ir dispersijų vektoriai bei perėjimų tikimybių reikšmės. HTK sistemoje modelių failai turi labai griežtai apibrėžtą struktūrą



Čia vaizduojami visų 25 balso komandų hmm failai.



Įrašai ir .mmf failai.

```
namespace Demo
{
    public partial class Demo : Form
    {
        private SpeechRecognitionEngine recognizer = new SpeechRecognitionEngine();

        public Demo()
        {
            InitializeComponent();

        }

        private void Start_Click(object sender, EventArgs e)
        {
    recognizer.SetInputToDefaultAudioDevice();
    label2.Text = "Sakykite";
            recognizer.SpeechRecognized += new EventHandler<SpeechRecognizedEventArgs>(recognizer_SpeechRecognized);
            recognizer.RecognizeAsync(RecognizeMode.Multiple);
        }

        void recognizer_SpeechRecognized(object sender, SpeechRecognizedEventArgs e)
        {

            string word = e.Result.Text.ToString();
            textBox1.Text = word;
        }



        private void Stop_Click(object sender, EventArgs e)
        {
            recognizer.RecognizeAsyncStop();
            richTextBox1.Clear();
            textBox1.Clear();
            label2.Text = "Pasirinkite sąrašą";
```

file length : 1761 lines : 67 Ln : 44 Col : 10 Sel : 0 | 0 Dos\Windows UTF-8

File Edit Search View Encoding Language Settings Macro Run Plugins Window ?

Irasai.TXT Form1.cs

```
31              recognizer.RecognizeAsync(RecognizeMode.Multiple);
32          }
33
34          void recognizer_SpeechRecognized(object sender, SpeechRecognizedEventArgs e)
35          {
36
37              string word = e.Result.Text.ToString();
38              textBox1.Text = word;
39          }
40
41
42
43          private void Stop_Click(object sender, EventArgs e)
44          {
45              recognizer.RecognizeAsyncStop();
46              richTextBox1.Clear();
47              textBox1.Clear();
48              label2.Text = "Pasirinkite sąrašą";
49              recognizer.UnloadAllGrammars();
50          }
51
52
53
54          private void Vaistai_Click(object sender, EventArgs e)
55          {
56  Grammar gramatika = new Grammar("vaistai.grxml", "Rule");
57              recognizer.LoadGrammar(gramatika);
58              label2.Text = "Spauskite Start";
59              richTextBox1.Clear();
60              richTextBox1.LoadFile("vaistai.rtf");
61          }
62
63
64          }
65      }
```

source file length : 1761 lines : 67 Ln : 44 Col : 10 Sel : 0 | 0 Dos\Windows UTF-8