ŠIAULIŲ UNIVERSITETAS
INFORMATIKOS, MATEMATIKOS IR E. STUDIJŲ INSTITUTAS
INFORMATIKOS KATEDRA

Jūratė Vaičiulytė

Informatikos magistrantūros specialybės II kurso nuolatinių studijų studentė

Automatinio šnekos atpažinimo metodų tyrimas ir taikymai balso įrašams stenografuoti

The Investigation of Automatic Speech Recognition Methods and Their Applications in the Stenography of Voice Records

MAGISTRO DARBAS

Darbo vadovas: Doc. dr. G. Felinskas
Recenzentas: Doc. dr. K. Žilinskas

Šiauliai, 2015

„Tvirtinu, jog darbe pateikta medžiaga nėra plagijuota ir paruošta naudojant literatūros sąraše pateiktus informacinius šaltinius bei savo tyrimų duomenis“

Darbo autoriaus _____
(vardas, pavardė, parašas)

Darbo tikslai ir uždaviniai

Tikslas – išanalizuoti automatinio šnekos atpažinimo (AŠA) metodus ir algoritmus bei pritaikyti juos kuriant programinio įrankio prototipą, skirtą garso įrašams stenografuoti.

Uždaviniai:

- Susipažinti su įvairiais automatinio šnekos atpažinimo metodais;
- Išanalizuoti mokslo darbus apie lietuvių kalbos automatinį šnekos atpažinimą;
- Suprojektuoti automatinio šnekos atpažinimo modelį, taikant pasirinktus AŠA metodus;
- Apmokyti akustinius modelius ir ištestuoti jų atpažinimo tikslumą;
- Sukurti programinio įrankio prototipą, skirtą garso įrašams stenografuoti;
- Atlikti sukurto programinio įrankio testavimą bei įvertinti atpažinimo tikslumą.

Darbo vadovo _____
(vardas, pavardė, parašas)

TURINYS

1 ĮVADAS

Kalba – tai natūraliausia ir lengviausia žmonių bendravimo forma. Automatinės šnekos atpažinimo (*angl. Automatic Speech Recognition*) technologijos leidžia žmonėms komunikuoti su kompiuterine technika ir kitais elektroniniais įtaisais. Automatinės šnekos atpažinimo sistemos identifikuoja žodžius, kuriuos sako žmogus, ir konvertuoja juos į tekstą. Šios sistemos apima įvairias sritis, tokias kaip: balsu valdomos vartotojo sąsajos, diktavimo įrankiai, automatinis telefono skambučių apdorojimas, automatinis balso vertimas iš vienos kalbos į kitą, stenografavimas ir t.t. [1,16] Šios technologijos gali būti naudojamos švietime, priemonių neįgaliesiems kūrime, taip pat buities ir darbo palengvinimo tikslais, o automatinės šnekos atpažinimo tyrimai palengvina jų tobulinimą.

Automatinio šnekos atpažinimo sistemos pritaikytos tik plačiai vartojamoms kalboms, tokioms kaip: anglų, japonų, prancūzų, vokiečių ir kt. Lietuvių kalbos struktūra yra gana sudėtinga, jai netinka kitoms kalboms sukurti akustiniai modeliai. Todėl yra svarbu kurti lietuvių kalbos atpažinimo sistemas, kad šios kalbos vartojimas nebūtų išstumtas iš modernių technologijų (pvz.: Vilniaus universiteto Matematikos ir informatikos Institute sukurtas atskirų lietuvių kalbos žodžių atpažinimo sistemos prototipas „Žodžių atpažintuvas“, Vilniaus Gedimino technikos universitete vykdomi lietuvių kalbos pavienių žodžių atpažinimo algoritmo įgyvendinimas, KTU universitete tiriamos užsienio kalbų automatinio šnekos atpažinimo variklių galimybės atpažinti lietuvių kalbos balso komandas ir t.t.) [3,13,7].

Automatinio šnekos atpažinimo sistemas, būtų galima suskirstyti į dvi pagrindines kategorijas: pirmoji apimtų programas, kurios gali interpretuoti atpažinimo rezultatus (komandų ir kontrolės programa), kita kategorija apimtų programas, kurios atpažinto teksto neinterpretuoja (diktavimo programa). [9] Balso įrašų stenografavimo įrankis priklausytų pastarajai kategorijai. Tokius įrankius dažnai naudoja daktarai, žurnalistai, kai reikia balsu sakomas pastabas versti į tekstinius failus duomenų išsaugojimui. [12]

Šio darbo **tikslas** – išanalizuoti automatinio šnekos atpažinimo (AŠA) metodus ir algoritmus bei pritaikyti juos kuriant programinio įrankio prototipą, skirtą garso įrašams stenografuoti.

Uždaviniai:

- Susipažinti su įvairiais automatinio šnekos atpažinimo metodais;
- Išanalizuoti mokslo darbus apie lietuvių kalbos automatinį šnekos atpažinimą;

- Suprojektuoti automatinio šnekos atpažinimo modelį, taikant pasirinktus AŠA metodus;
- Apmokyti akustinius modelius ir ištestuoti jų atpažinimo tikslumą;
- Sukurti programinio įrankio prototipą, skirtą garso įrašams stenografuoti;
- Atlikti sukurto programinio įrankio testavimą bei įvertinti atpažinimo tikslumą.

2 ANALITINĖ DALIS

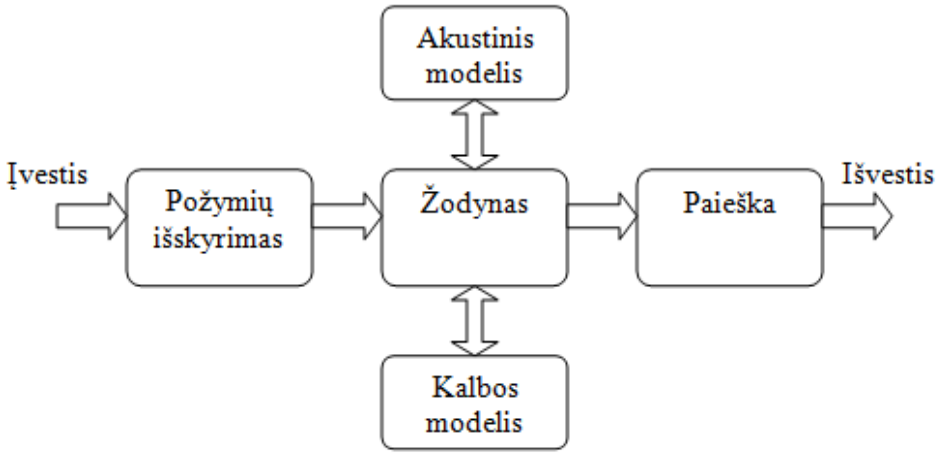
2.1 Automatinis šnekos atpažinimas

2.1.1 Automatinio šnekos atpažinimo veikimo principas

Automatinio šnekos atpažinimo sistemos gautą įvestį – garso signalą – apdoroja ir išveda tekstinį įvesties atitikmenį (*žr. 1 pav.*). Matematiškai automatinės atpažinimo sistemos uždavinį būtų galima suformuluoti taip: turint garso signalo požymių seką $X = x_1, x_2, \ldots, x_n$ reikia rasti žodžių seką $W = w_1, w_2, \ldots, w_m$, kuri turi maksimalią *posterior* tikimybę *P(W|X)*, išreikštą Bajeso formule:

$$W = \arg\max_{w} P(W \mid X) = \arg\max_{w} \frac{P(W)P(X \mid W)}{P(X)} \qquad (1)$$

Čia *P(X|W)* yra tikimybė, kad tariant žodžių seką *W*, bus stebima požymių seka *X*, *P(W)* – *a priori* tikimybė, kad bus ištarta žodžių seka *W*, *P(X)* – tikimybė, kad bus stebima požymių seka *X*. [1,16]



1 pav. Šnekos atpažinimo sistemos struktūra

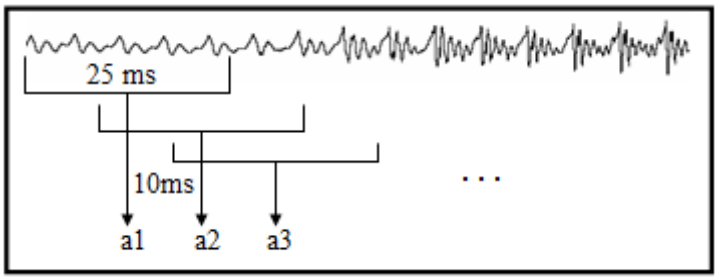
Automatinis šnekos atpažinimas paprastai susideda iš dviejų etapų – pirminio apdorojimo ir galutinio apdorojimo. Pirminis apdorojimas apima požymių išskyrimą, o galutinis apdorojimas sudaro šnekos atpažinimo variklį, kuris turi akustinį modelį, žodyną ir gramatiką. Jei visos šios dalys yra korektiškos, šnekos atpažinimo variklis identifikuoja labiausiai tikėtiną atitikmenį gautai įvesčiai ir grąžina atpažintus žodžius tekstu. Tinkamų požymių išskyrimo ir šnekos atpažinimo metodų parinkimas turi didelės įtakos atpažinimo sistemos tikslumui. Įvairūs požymių išskyrimo ir šnekos atpažinimo metodai yra aprašyti kituose skyriuose. [1, 15, 16]

2.1.2 Šnekos požymių išskyrimo metodai

Skaitmeninis signalas $Y = y_1, y_2, \ldots, y_k$ į automatinio šnekos atpažinimo sistemą dažniausiai įvedamas iš failo arba mikrofono. Šiame garso signale be ištartų žodžių gali būti daug kitos informacijos, pvz.: aplinkos triukšmas, akcentas, intonacija ir t.t. Požymių išskyrimo dalies užduotis yra tansformuoti garso signalą *Y* į požymių seką *X*. Požymių seka turėtų atitikti tokius kriterijus: [15]

- Lengvai išmatuojami šnekos požymiai;
- Garsų klasės būtų kuo kompaktiškesnės požymių erdvėje;
- Požymiai kuo mažiau priklausytų nuo įrašymo aplinkos, kalbėtojo, kalbėjimo, t.y. turėtų kuo mažiau šnekos atpažinimui nereikalingos informacijos.

Požymių išskyrimas dažniausiai remiasi dažnine signalų analize. Daroma prielaida, kad šnekos signalas yra stacionarus trumpame intervale, tad signalas *Y* skaidomas mažais persidengiančiais langais (kadrais). Dažniausiai naudojamas nuo 15ms iki 30ms trukmės langas ir požymių išskyrimo metu langas slenkamas po 10–20ms (*žr. 2 pav.*). Naudojantis Furje transformacija arba tiesine prognoze, kiekvienam langui randamas dažnių spektras. Langą atitinkančių požymių seka x_i gaunama atliekant netiesines transformacijas (pavyzdžiui, dažnių skalės iškraipymo transformacija, logaritmavimas). Tam, kad būtų įvertinti pasikeitimai signale, naudojamos požymių išvestinės, skaičiuojamos tarp gretimų langų. Dažniausiai į požymių vektorių įtraukiamos pirmosios ir antrosios eilės išvestinės. [15]

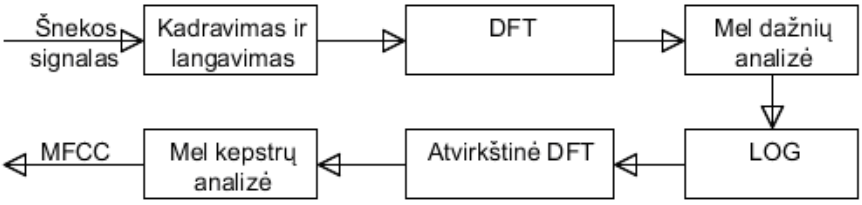


2 pav. Signalo skaidymas langais (kadrais) [2]

Mel dažnių kepstro koeficientai *(angl. Mel Frequency Cepstral Coefficients (MFCC))*

MFCC (*žr. 3 pav.*) metodu išskirti požymiai yra dažnai naudojami šnekos atpažinime. Šis metodas remiasi trumpalaike (angl. *short-term analysis*) analize, taip iš kiekvieno kadro yra apskaičiuojamas MFCC vektorius. Pirmiausia garso signalui atliekamas langų (kadrų) išskyrimas, kad sumažintų garso signalo pertrūkius. Tuomet atliekama Furje transformacija

(angl. *DFT*) ir taip sugeneruojamas Mel filtras (*angl. Mel filter bank*). Galiausiai atliekama atvirkštinė Furjė transformacija ir apskaičiuojami Kepstro koeficientai. [16]



3 pav. Signalo požymių išskyrimas MFCC metodu [16]

Pavyzdžiui, jei lango dydis pasirenkamas 25ms ir paslenkama po 10ms gali būti išgaunami 39 požymiai iš 10ms lango: [2]

- Absoliutūs koeficientai: Log Frame Energy (1) ir MFCC (12);
- Pirmos eilės išvestinės iš 13 absoliučių koeficientų;
- Antros eilės išvestinės iš 13 absoliučių koeficientų.

Apie kitus požymių išskyrimo metodus, tokius kaip Tiesinio prognozavimo metodą ir Tiesinės suvokimo prognozės modelio analizę žiūrėti 3 priede.

2.1.3 Akustinis modelis

Akustinio modelio užduotis yra įvertinti žodžių sekos tikimybę $P(X|W)$ formulei. Teoriškai būtų galima surinkti daug žodžio w garso pavyzdžių ir taip sudaryti tikimybinio požymių vektorių pasiskirstymo priklausomybę nuo žodžio, bet praktiškai tai sunkiai įgyvendinama, nes didelio žodyno atveju daug žodžių retai pasitaiko mokymo duomenyse arba iš viso net neturima žodį atitinkančio garso įrašo. Todėl dažniausiai požymių vektoriaus X skirstinys modeliuojamas pagal mažesnius negu žodis fonetinius vienetus – fonemas, kontekstines fonemas arba skiemenis (fonemu, kontekstinių fonemų paaiškinimus). [15]

Šiam skirstiniui modeliuoti naudojami paslėptieji Markovo modeliai (toliau PMM). PMM galima įsivaizduoti kaip atsitiktinį procesą, kuris keliauja per būsenų aibę S ir generuoja požymių vektorius X. Formaliai PMM yra aprašomas: [15]

- Būsenų aibe: $S = 1,...,N$.
- Perėjimo tikimybių matrica: $A = [a_{ij}],\ 1 \le i, j \le N$. a_{ij} žymi perėjimo tikimybę iš būsenos i į būseną j: $a_{ij} = P(s_t = j | s_{t-1} = i)$.
- Generuojamų požymių skirstinių funkcijomis: $B = [b_j],\ 1 \le j \le N$.
  Čia $b_j(x_t) = P(x_t | s_t = j)$ – tikimybė, kad būsena j sugeneruos požymių vektoriu x_t.

Kiekvienas akustinis vienetas modeliuojamas vienu PMM, kuris būna sudarytas iš kelių būsenų. Dažniausiai naudojami trijų būsenų (garso pradžia, vidurys ir pabaiga) PMM. Kad būtų patogiau sujungti atskirus PMM tarpusavyje, prie trijų būsenų pridedamos dar dvi negeneruojančios požymių vektoriaus būsenos. Konkretaus žodžio PMM tinklas gaunamas sujungiant akustinių vienetų, atitinkančių žodžio tarimą, PMM. Taip gauti žodžių PMM tinklai gali būti jungiami toliau suformuojant žodžių sekų PMM tinklus. [15]

Bendruoju atveju nežinoma, kokia būsenų seka sugeneravo požymių vektorių. Egzistuoja iteracinis algoritmas *Forward-Backward*, leidžiantis efektyviai apskaičiuoti šią tikimybę. Didelio žodyno atveju formuojamas statinis arba dinaminis žodžių tinklas, sudarytas iš daugelio PMM, ir tinkle ieškoma tokia būsenų seka S, kuri sugeneruoja požymių vektorius X su didžiausia tikimybe. Geriausiai sekai surasti naudojamas Viterbi algoritmas. [15]

2.1.4 Kalbos modelis

Nedidelio žodyno atskirai tariamų žodžių atpažinimo atveju kartais kalbos modelis (toliau KM) nenaudojamas ir pasikliaujama tik akustiniu modeliu, bet rišlios šnekos atpažinimo sistemose KM būtinas. Visų pirma dėl to, kad šnekoje egzistuoja taisyklės (gramatika), kaip žodžiai jungiami vienas su kitu ir šių žinių panaudojimas leidžia labai sumažinti klaidingai atpažįstamų žodžių kiekį. Žodžiai gali skambėti panašiai arba netgi vienodai ir tokiu atveju tik naudojant lingvistines žinias, kurios sukaupiamos KM, galima priimti teisingą sprendimą. [15]

Kalbos modelis susideda iš dviejų dalių:

- žodyno,
- gramatikos taisyklių, kaip žodžiai gali būti rikiuojami ir taip sudaro frazes, sakinius.

KM gramatika gali būti aprašoma formaliomis lingvistinėmis taisyklėmis, pavyzdžiui, bekontekstėmis gramatikomis, arba sudaroma remiantis statistiniais metodais, t.y. naudojant statistinius KM, kurie modeliuoja žodžio pasirodymo tekste tikimybinius skirstinius pagal gretimus žodžius. Bekontekstės gramatikos gali būti naudojamos tik specializuotiems šnekos atpažinimo uždaviniams spręsti. Tokias gramatikas tiesiogiai panaudoti didelio žodyno atpažinimo sistemose neįmanoma, nes praktiškai nėra galimybės apibrėžti visų taisyklių, esančių šnekoje. [15]

Tokių trūkumų neturi statistiniai KM. Be to, statistiniai KM plačiai naudojami rišlios šnekos atpažinimo sistemose ir dėl jų paprastumo. Statistinis KM kiekvienai žodžių sekai W turi priskirti tam tikrą tikimybę $P(W)$. [15]

Tarkim, $W = w_1, w_2, ..., w_K$ ($w_i \in V$) yra K žodžių seka, kurios tikimybę norima įvertinti. Tada pagal Bajeso taisyklę $P(W)$ formaliai gali būti užrašoma taip:

$$P(W) = \prod_{i=1}^{K} P(w_i \mid w_1, ..., w_{i-1}) = \prod_{i=1}^{K} P(w_i \mid h_i) \qquad (2)$$

Čia $P(w_i|w_1, ..., w_{i-1})$ yra tikimybė, kad bus ištartas žodis w_i, jei prieš tai buvo pasakyti $w_1, ..., w_{i-1}$. h_i bus vadinama žodžio w_i istorija. [15]

Taigi KM turi įvertinti visas galimas tikimybes $P(w_i|h_i)$. Kalbos modelio tikimybės įverčiai surandami naudojantis tekstynu. Kuo tekstynas yra didesnis, tuo patikimesni yra tikimybių įverčiai. Deja, ir neilgoms žodžių sekoms (kai i mažas), galimų žodžio istorijų $w_1, ..., w_{i-1}$ yra labai daug. Net ir turint labai didelį tekstyną sunku patikimai įvertinti sekų tikimybes. Problema sprendžiama grupuojant žodžio istorijas į ekvivalenčias klases. [15]

Kalbos modeliavimo užduotis yra parinkti kuo geresnę istorijos klasifikavimo funkciją. Funkcija turi leisti ir gerai prognozuoti būsimą žodį, ir klasės turi būti gana dažnos, kad būtų galima patikimai įvertinti tikimybes. [15]

Populiariausi KM yra n-gramos. Daroma prielaida, kad žodis priklauso tik nuo $n-1$ ankstesnio žodžio. Klasifikavimo funkcija šiuo atveju labai paprasta: [15]

$$\Phi(w_1, ..., w_{i-1}) = w_{i-n+1}, w_{i-n+2}, ..., w_{i-1} \qquad (3)$$

Čia n žymi modelio eilę. Tada formulė sekos tikimybei apskaičiuoti n-gramų atveju yra tokia:

$$P(W) = \prod_{i=1}^{K} P(w_i \mid w_{i-n+1}, ..., w_{i-1}) . \qquad (4)$$

Nors ir paprastos, bet n-gramos sugeba įvertinti ir sintaksinius, ir prasminius žodžių ryšius, todėl yra vieni iš geriausių modelių. Tyrimuose n-gramos paprastai laikomos atskaitos modeliais ir koks nors modeliavimo patobulinimas įvertinamas lyginant su n-gramomis. Įvairūs sudėtiniai KM variantai gali pagerinti n-gramų modelius, tačiau beveik visais atvejais pagerinimas pasiekiamas tik tuo atveju, jei n-grama yra naudojama kartu. Bigramos ($n = 2$) ir trigramos ($n = 3$) atsižvelgia į žodžio istoriją. [15]

2.1.5 Paieška

Atpažinimo sistemoje akustinis modelis sudaromas iš atskirų PMM, kurie atitinka bazinius akustinius vienetus, dažniausiai fonemas arba kontekstines fonemas. Akustinių vienetų PMM sujungiami naudojantis tarimo žodynu, taip suformuojant akustinius žodžių modelius.

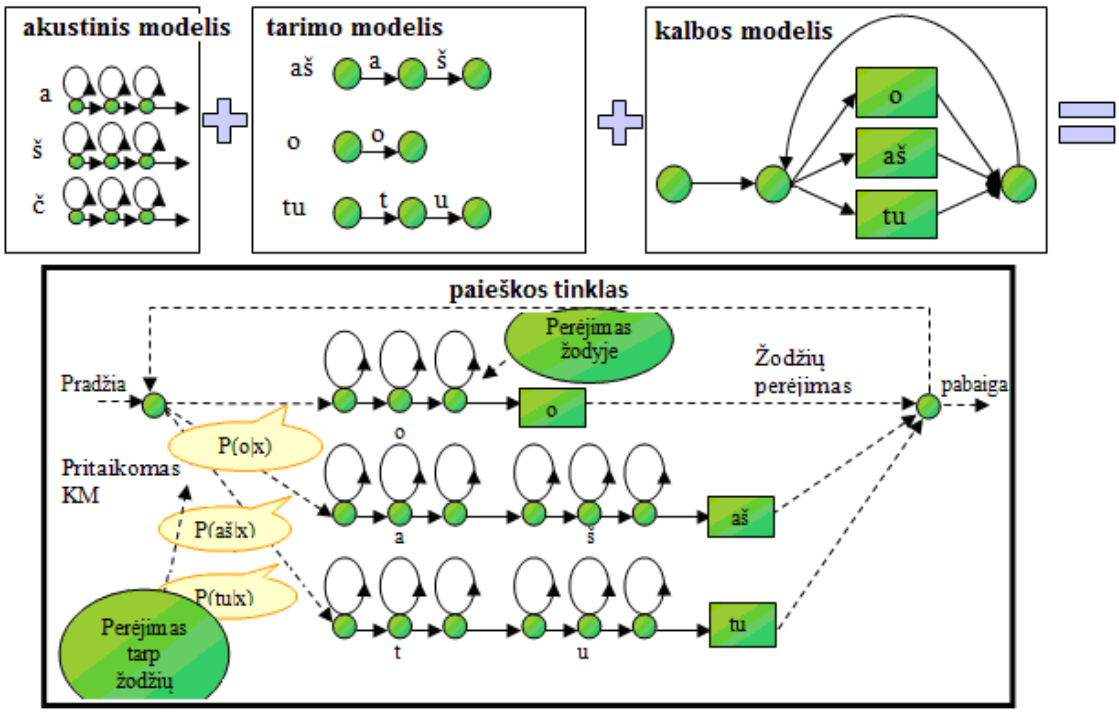
Kadangi reikia atpažinti ir rišlios šnekos žodžių sekas, žodžių akustiniai modeliai taip pat turi buti sujungiami į bendrą tinklą. [15]

Šio tipo paieškos tinklas vadinamas statiniu, nes tinklas suformuojamas kuriant sistemą ir paieškos metu nekinta (*žr. 4 pav.*). Tinklą galima įsivaizduoti kaip baigtinių būsenų automatą. KM įjungiamas į tinklą naudojant perėjimų tarp busenų, atitinkančių žodžio pradžią ir pabaigą, svorius. Tinklo sudetingumą apibrėžia KM. Paprastai statinis paieškos tinklas naudoja bigramos, trigramos arba dar aukštesnės eilės modelius. [15]

Galima į tinklą įjungti ir klasių n-gramos modelį. Paieška tokiame tinkle atliekama su modifikuotu Viterbi algoritmu, kuris naudoja spindulio paiešką, t.y. stengiamasi sumažinti paieškos erdvę atmetant neperspektyvius kelius tinkle. Praktikoje Viterbi paieškai žodžių tinkle realizuoti dažniausiai naudojamas žetono siuntimo principas. [15]

Viterbi paieška. Žetono siuntimo algoritmas: [2]

- Inicializuoti visas būsenas su žetonu, kurio istorija tuščia ir tikimybe, kad tai bus pradinė būsena;
- Kiekvienam kadrui a_k
  - Kiekvienam žetonui t būsenoje s su tikimybe $P(t)$, istorija H
    - Kiekvienai būsenai r
      - Pridėti naują žetoną būsenai s su tikimybe $P(t)\ P_{s,r}\ P_r(a_k)$, ir istorija $s.H$



4 pav. Paieškos tinklas [2]

2.1.6 Šnekos atpažinimo sistemos tikslumas

Šnekos atpažinimo sistemos paprastai apibūdinama nurodant atpažinimo tikslumą ir vykdymo greitį. Atpažinimo tikslumas gali būti matuojamas žodžių klaidų lygiu (angl. *Word Error Rate (WER)*). Dažniausiai matavimas atliekamas su testine imtimi. [8]

Pirmiausia sistemos atpažinta žodžių seka yra sulyginama su tikrąja žodžių seka. Tuomet *WER* gali būti apskaičiuojamas pagal formulę:

$$WER = \frac{S + D + I}{N} \times 100\% \qquad (5)$$

kur S – kiekis klaidingai atpažintų žodžių , D – kiekis praleistų žodžių, I – kiekis įterptų žodžių. N – žodžių kiekis testinėje imtyje.

Tikslumą vertinti galima pagal tai, kokia dalį žodžių sistema atpažįsta teisingai (angl. *Word Recognition Rate (WRR)*). Tai galima apskaičiuoti pagal formulę:

$$WRR = 1 - WER = \frac{N - S - D - I}{N} \times 100\% = \frac{H - I}{N} \times 100\% \qquad (6)$$

kur *H* yra *N-(S+D)* – teisingai atpažintų žodžių kiekis. [8]

2.2 Automatinio šnekos atpažinimo sistemų klasifikavimas

AŠA sistemos gali būti klasifikuojamos pagal tai, koks yra šnekėjimo tipas, kalbėtojo modelio tipas, žodyno tipas. [16] Automatinio šnekos atpažinimo sistemos gali būti skirstomos į grupes **pagal šnekėjimo pobūdį**: [1,8]

- *Pavienių žodžių atpažinimas:* šios sistemos atpažįsta tariamus pavienius žodžius, atskirtus tyla. Šios sistemos turi „Klausymo/Neklausymo“ būsenas, per kurias vartotojas turi laukti (paprastai per šias pauzes yra vykdomas apdorojimas).[1,8] Tokios sistemos naudingos, kai vartotojas turi ištarti pavienius žodžius, komandas. Šias sistemas lengviausia realizuoti, kadangi žodžių ribos yra akivaizdžios, o patys žodžiais aiškiai ištariami.[16]
- *Rišlių frazių atpažinimas:* šios sistemos panašios į pavienių žodžių atpažinimo sistemas, tačiau jos apdoroja žodžius, pasakytus su minimalia pauze tarp jų. [1,8]
- *Ištisinės šnekos atpažinimas:* šios sistemos leidžia vartotojui kalbėti beveik natūraliai (tai diktavimas kompiuteriui). [1,8] Ištisinės šnekos atpažinimo sistemas sunkiausia realizuoti, nes jos turi nustatyti žodžių ribas. Padidinus žodyną, atpažinimo problema tampa sudetingesnė, nes norint atpažinti žodį reikia rinktis iš daugelio variantų, kurie gali buti akustiškai panašūs. Klaidos tikimybė padidėja didėjant žodynui. [16]

- *Spontaniškos šnekos atpažinimas:* šios sistemos atpažįsta natūralią šneką. Tokios sistemos gali apdoroti šneką, kuri pasižymi įvairiu kiekiu savybių (beprasmiais garsais, pvz.:„uhm“, „mhm“, tempo ir balso kitimu, vartojamu žodynu, pauzėmis). [1,8]

Kalbėtojo modelio tipas:

- *Nuo kalbėtojo priklausomos* sistemos atpažįsta ją apmokiusio asmens žodžius. Šios sistemos tikslesnės tam tikram vartotojui, tačiau mažiau tikslios kitiems kalbėtojams. [16]
- *Nuo kalbėtojo nepriklausomos* sistemos gali atpažinti įvairių kalbėtojų žodžius, jos nereikalauja individualaus apmokymo ir skirtingų asmenų šnekai atpažinti naudoja tą patį etalonų rinkinį. [16]

Žodyno tipai.

Automatinio šnekos atpažinimo sistemų naudojami žodynai gali būti klasifikuojami į tokias grupes: mažas žodynas – dešimtys žodžių, vidutinis žodynas – šimtai žodžių, didelis žodynas – tūkstančiai žodžių, labai didelis žodynas – dešimtys tūkstančių žodžių ir daugiau. [16]

2.3 Automatinio šnekos atpažinimo metodai

Ankstesniais metais dinaminio programavimo metodai buvo taikomi siekiant išspręsti šnekos atpažinimo problemas. Vėlesni tyrimai buvo grindžiami dirbtinių neuroninių tinklų metodais, kuriuose lygiagretūs skaičiavimai mėgdžioja biologines neuronines sistemas. Visai neseniai stochastinio modeliavimo metodai buvo įtraukti sprendžiant šnekos atpažinimo problemas, pvz.: paslėptieji Markovo modeliai. Šiuo metu daug naujų tyrimų apima ištisinės/spontaniškos šnekos atpažinimą, naudojant labai didelius žodynus, PMM, DNT ar hibridinius metodus. Šie metodai yra trumpai paaiškinti žemiau. [16]

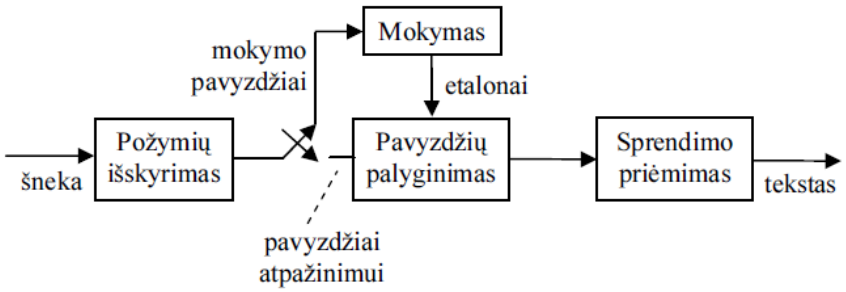
2.3.1 Akustinis fonetinis metodas (angl. *Acoustic phonetic approach*)

Šis metodas grindžiamas kalbos garsų paieška ir jų susiejimu su atitinkamomis žymėmis (etiketėmis). Akustinis fonetinis metodas teigia, kad šnekoje egzistuoja baigtiniai skiriamieji fonetiniai vienetai (fonemos), kurie gali būti aprašomi akustiniais savybių rinkiniais, išgautais iš šnekos garso signalo. Pirmasis akustinio fonetinio metodo žingsnis yra spektrinė šnekos analizė kartu su požymių nustatymu, kuris konvertuoja spektrinius matavimus į savybių rinkinius, kurie aprašo fonetinių vienetų bendrąsias akustines savybes. Kitas žingsnis yra segmentavimo ir ženklinimo etiketėmis etapas, kuriame šnekos signalas yra suskirstomas į statinius akustinius segmentus, prie kiekvieno iš jų priskiriant po vieną ar daugiau fonetinių etikečių (tokiu būdu yra charakterizuojama šneka). Paskutiniu šio metodo žingsniu siekiama nustatyti teisingą žodį (ar

keletą žodžių) iš fonetinių etikečių sekų. Į validavimo etapą yra įtraukiami ir kalbiniai (lingvistiniai) apribojimai (pvz.: žodynas, sintaksė ir kitos semantinės taisyklės). [1,8]

2.3.2 Struktūrų atpažinimo metodas (angl. *Pattern Recognition approach*)

Struktūrų atpažinimo metodas turi du pagrindinius žingsnius (*žr. 5 pav.*): struktūrų apmokymas (*angl. pattern training*) ir struktūrų palyginimas (*angl. pattern comparison*). Mokymo procedūros metu vyksta akustinių struktūrų (modelių) kūrimo procesas, mokymas. Atpažinimo etape nežinomas ištarimas lyginamas su visais modeliais ar jų kombinacijomis ir atrenkamas geriausias modelis tam tikro atstumo prasme. Šnekos struktūrų vienetai gali būti aprašomi šnekos šablonais ar statistiniais modeliais (pvz., PMM) ir taikomi garsui (vienetas mažesnis už žodį), žodžiui ar frazei. [1,8,10]



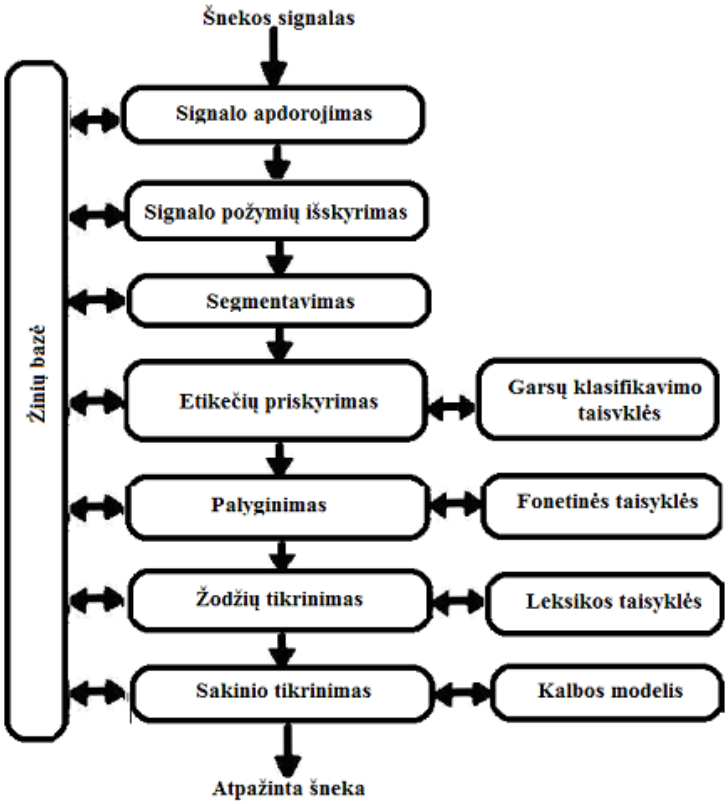
5 pav. Struktūrų atpažinimo metodo schema [10]

2.3.2.1 Šablonais grįsti metodai (angl. *Template-Based Approaches*)

Šio proceso metu nežinoma šneka (ištarti žodžiai) yra lyginama su iš anksto įrašytais žodžiais (šablonais), siekiant rasti labiausiai panašų atitikmenį. Paprastai yra sudaromi viso žodžio šablonai. Tai didelis privalumas, nes galima išvengti klaidų dėl mažesnių akustinių vienetų, tokių kaip fonemų, klasifikavimo ar segmentavimo. Viena iš pagrindinių idėjų yra gauti šnekos kadrų sekas (kiekvienam žodžiui) per tam tikrą vidurkinimo procedūrą ir remtis lokaliais spektriniais atstumo matavimais šablonų palyginimui. Kita svarbi idėja yra naudoti tam tikrą dinaminio programavimo formą, kad būtų galima laikinai palyginti šablonus, siekiant atsižvelgti į kalbėjimo tempų skirtumus (tarp kelių kalbėtojų ir vieno kalbėtojo, kartojančio tą patį žodį). Tačiau šis metodas turi ir trūkumų – iš anksto įrašyti šablonai yra fiksuoti, todėl pokyčiai (*angl. variations*) šnekoje gali būti modeliuojami naudojantis tik daugeliu vieno žodžio šablonais, o tai būtų nepraktiška. Šablonų parengimas ir lyginimas tampa per brangus, nes žodyno dydis padidėja daug daugiau nei keli šimtai žodžių. [1,8,16]

2.3.3 Žiniomis grįsti metodai (angl. *Knowledge-Based Approaches*)

„Ekspertų“ žinios apie svyravimus šnekoje yra ranka koduojami (angl. *hand-coded*) į sistemą. Panaudojant šnekos signalo savybių rinkinį apmokymo sistema automatiškai iš pavyzdžių sugeneruoja taisyklių rinkinį. Šios taisyklės yra gaunamos iš parametrų, kurie suteikia daugiausia informacijos apie klasifikavimą. Šnekos atpažinimas atliekamas kadro lygmeniu, kuomet yra sudaromas sprendimų medis ir klasifikuojamos taisyklės (*žr. 6 pav.*). Šis metodas turi privalumų – aiškiai modeliuojami svyravimai šnekoje. Tačiau tokias „ekspertų“ žinias sunku išgauti ir sėkmingai panaudoti, todėl šis metodas buvo įvertintas kaip nepraktiškas. [8,16]



6 pav. Žiniomis grįstų metodų šnekos atpažinimo schema

2.3.4 Neuroniniais tinklais grįsti metodai (angl. *Neural Network-Based Approaches*)

Dirbtiniais neuroniniais tinklais (DNT) grįstas metodas kalbai atpažinti pradėtas taikyti praėjusio amžiaus devintajame dešimtmetyje ir yra jauniausias iš nagrinėjamų metodų. Iš pradžių neuronų tinklai taikyti fonemoms, skiemenims atpažinti, vėliau atskiriems žodžiams ir ištisinei kalbai. Jis geba išspręsti daug sudėtingesnes atpažinimo užduotis, tačiau parodo prastesnius rezultatus nei paslėpti Markovo modeliai (PMM), kai atpažinimui naudojami labai dideli žodynai. [5,16] Plačiau apie neuroninius tinklus automatiniame šnekos atpažinime skaitykite 2 priede.

2.3.5 Dinaminiu laiko skalės iškreipimu grįsti metodai (angl. *Dynamic Time Warping (DTW)-Based Approaches*)

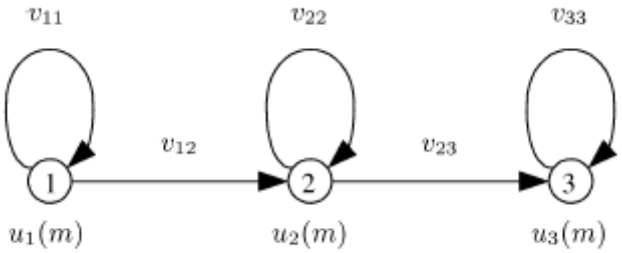
Dinaminis laiko skalės iškreipimo algoritmas yra skirtas matuoti panašumą tarp dviejų sekų, kurios gali skirtis trukme ar greičiu. Automatinės šnekos atpažinimo sistemose šis metodas susidorodavo su įvairiu kalbėjimo greičiu. Jis leidžia kompiuteriui rasti optimalų dviejų gautų sekų lyginimo rezultatą (pvz., laiko eilučių) su tam tikrais apribojimais. [1,16]

Betkokius duomenis, aprašomus tiesiškai, galima išanalizuoti DTW metodu. Šis sekų palyginimo metodas yra dažnai naudojamas paslėptų Markovo modelių kontekste. DTW algoritmas ypač tinka lyginimui sekų su trūkstama informacija, tačiau turi būti pateikiami pakankamai ilgi segmentai, kuriuose atsirastų atitikmenų. Optimizavimo procesas yra atliekamas naudojant dinaminį programavimą. Tam, kad surastų geriausią atitikmenį, kadras po kadro palyginant šabloną ir įvesties garso signalą DTW „ištęsia“, „suspaudžia“ įvairius garso signalo segmentus. DTW yra gana efektyvus pavienių žodžių atpažinime ir gali būti modifikuotas taip, kad galėtų atpažinti ir rišlias frazes. [1,8,16]

2.3.6 Paslėptaisiais Markovo Modeliais (PMM) grįstas šnekos atpažinimas (angl. *Hidden Markov Model-Based Speech Recognition)*

PMM yra populiarus metodas, nes jie gali būti apmokomi automatiškai ir juos galima paprastai panaudoti skaičiavimams. Viso žodžio PMM galima sukurti sujungiant atskirų fonemų (*angl. phone*) PMM, apskaičiuojant žodžio sekos tikimybes ir viso tinklo paieškos geriausio kelio, atitinkančio optimalią žodžio seką, suradimui. Šio modelio parametrai yra būsenų perėjimo tikimybės ir vidurkių, dispersijų svoriai, apibūdinantys būsenos išvesties skirstinius. Kiekvienas žodis ar fonema turės skirtingą išvesties skirstinį. PMM, skirtas keletos žodžių ar fonemų sekai, yra padaromas sujungiant individualiai atskiriems žodžiams ir fonemoms apmokytą PMM. [1,16]

PMM (*žr. 7 pav.*) panaudojimas šnekai atpažinti remiasi prielaida, kad kalbos signalas yra atsitiktinis procesas, kurio parametrus galima nustatyti. Šiame metode šnekos pavyzdžiai modeliuojami paslėptaisiais Markovo modeliais. [14]



7 pav. Trijų būsenų Markovo modelis [14]

Šiuose modeliuose nagrinėjamas dvigubas atsitiktinis procesas, kuriame vyksta du atsitiktiniai procesai: perėjimas iš vienos būsenos į kitą ir stebėjimas būsenoje. Dažniausiai naudojami pirmos eilės Markovo modeliai, kuriuose perėjimo tikimybė priklauso tik nuo ankstesnės būsenos. Paslėptuoju vadinamas antrasis procesas, kadangi jis vykdomas per pirmąjį procesą ir tiesiogiai nestebimas. Perėjimus nusako tikimybių v_{ij} aibė, m-ojo simbolio b_m

stebėjimą būsenoje i – tikimybių $u_i(b_m)$ aibė. Tokį pavyzdžio modelį nusako penki parametrai: M – stebėjimo simbolių skaičius būsenoje, T – būsenų skaičius, V – perėjimo tikimybių pasiskirstymas, U – stebėjimų būsenoje tikimybių pasiskirstymas ir π – pradinio buvimo būsenoje tikimybių pasiskirstymas. Toks modelis gali būti panaudotas kaip stebėjimų sekos $X = (x_1, x_2,...,x_T)$ generatorius: [14]

1. Laiko momentu *1* su tikimybe π_i patenkame į pradinę būseną $t_1=i$.
2. Būsenoje i su tikimybe $u_i(b_m)$ stebime m-ąjį simbolį $x_1 = u_i(b_m)$.
3. Laiko momentu *2* su tikimybe v_{ij} pereiname į būseną $t_2=j$.
4. Būsenoje j su tikimybe $u_i(b_m)$ stebime m-ąjį simbolį $x_2=u_i(b_m)$.
5. Žingsnius *3* ir *4* kartojame visiems $t_1,t_2,..,t_T$, taip gaudami stebėjimų vektorių.

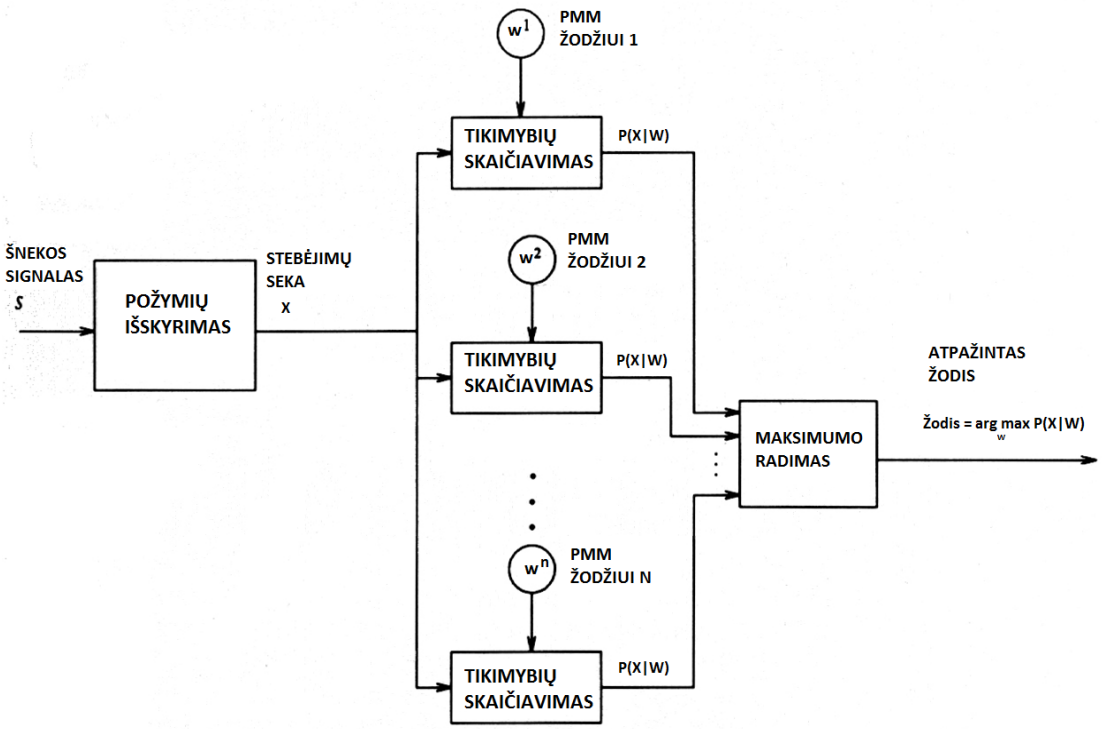
Šie penki dydžiai nusako konkretų paslėptąjį Markovo modelį, nors tam užtenka nusakyti rinkinį $\lambda=(V,U,\pi)$. Kadangi π yra konstanta visiems PMM, naudojamiems šnekai atpažinti, rinkinys paprastėja iki $\lambda=(V,U)$. [5]

Tarkime turime V šnekos pavyzdžių, kuriems atpažinti norime pritaikyti PMM metodą. Pirmasis žingsnis – žodyno sukūrimas. Kiekvienam iš V pavyzdžių sukuriame modelį λ. Modelio tikimybiniai parametrai V, U ir π nustatomi taikant įvertinimo procedūras iš apmokymui pateiktų pavyzdžių. Nagrinėjant nežinomąjį kalbos pavyzdį, mes atliekame signalo analizę, taip gaudami stebėjimų seką X. Atpažintuoju pavyzdžiu paskelbiamas etaloninis pavyzdys, kurio modelis geriausiai atitinka nagrinėjamą stebėjimų seką (*žr. 8 pav.*). Modelio atitikimą stebėjimų sekai įvertinant tikėtinumu, kad nagrinėjamoji stebėjimų seka yra sygeneruota modelio, atpažintuoju pavyzdžiu skelbiamas etalonas: [14]

$$Z = \arg\max_{1<k<V} P(X \mid \lambda_k) \qquad (7)$$

Taigi norint sėkmingai naudoti PMM kalbos signalams atpažinti reikia išspręsti tris uždavinius: [5]

- Įvertinimo uždavinį: turint stebėjimų seką $X=(x_1, x_2,...,x_T)$ ir grandinę aprašančio modelio parametrus $\lambda=(V,U,\pi)$, reikia apskaičiuoti tikimybę $P(X|\lambda)$, kad nagrinėjamoji stebėjimų seka buvo sugeneruota nagrinėjamo modelio;
- Paslėptųjų būsenų nustatymo uždavinį: turint stebėjimų seką $X=(x_1, x_2,...,x_T)$, reikia nustatyti būsenų seką, kuri būtų optimali tam tikro pasirinkto prasmingo kriterijaus prasme;
- Apmokymo uždavinį: kaip parinkti modelio parametrus, kad būtų maksimizuota tikimybė $P(O|\lambda)$.



8 pav. Automatinis šnekos atpažinimas naudojant PMM

2.4 Lietuvių kalbos automatinis šnekos atpažinimas

Lietuvoje kalbos atpažinimo darbai pradėti aštuntojo dešimtmečio pabaigoje – devintojo pradžioje. Pirmosiose šnekos atpažinimo sistemose buvo naudojamas dinaminės laiko skalės kraipymo metodas, realizuotas naudojant dinaminį programavimą. Devintojo dešimtmečio viduryje šnekai atpažinti pritaikyti paslėptieji Markovo modeliai. Signalui analizuoti buvo naudojama tiesinės prognozės, kepstrinė analizės, tačiau didžiausias atpažinimo tikslumas pasiektas naudojant melų skalės kepstro požymių sistemą. Vėliau šnekai atpažinti buvo pritaikyti dirbtiniai neuronų tinklai. Bandyta kurti ir hibridines atpažinimo sistemas jungiant PMM su DTW ir DNT, tačiau ypatingų rezultatų nepasiekta. [14]

Lietuvių šnekos atpažinimo akustinis modeliavimas. Buvo tirtas žodžiais, skiemenimis, kontekstiniais skiemenimis, fonemomis ir kontekstinėmis fonemomis grįstas šnekos atpažinimas. Tyrimai atlikti izoliuotiems žodžiams ir ištisinei šnekai. [10]

Lietuvių kalbos pavienių žodžių atpažinimo algoritmo įgyvendinimas lauku programuojama logine matrica (LPLM). Kalbos požymiams išskirti buvo taikoma kepstrinė šnekos analizė. Požymiams palyginti taikomas dinaminis laiko skalės kraipymo (DSLK) metodas. Sudaryta 100 žodžių požymių biblioteka. Pasiektas 94 % atpažinimo tikslumas priklausomai nuo kalbėtojo ir 58 % – nepriklausomai nuo kalbėtojo. [13]

Statistiniai kalbos modeliavimo (standartinių n-gramų, klasių modelių, nuotolinių bigramų, netolimosios praeities, žanrų mišinių, morfologinių modelių) metodai įvertinti ir palyginti taikant juos lietuvių kalbai modeliuoti. Eksperimentiniai tyrimai buvo atliekami su 84 mln. žodžių apimties tekstynu. Sukurtas automatinio lietuvių rišlios šnekos atpažinimo sistemos prototipas, kurios veikimo tikslumas buvo įvertintas sprendžiant bendro pobūdžio diktavimo sistemos (priklausomos nuo kalbetojo) uždavinį. Tyrimai buvo atliekami naudojant daugelio diktorių, 21 valandos trukmės garsyną; naudojant 1,15 mln. žodžių tarimo žodyną bei sudėtinį klasių kalbos modelį, gautas žodžių atpažinimo tikslumas siekė 73,72 %. [15]

Lietuvių kalbos atpažinimui yra sukurtos kelios programos. Pirmoji programa *„Atpažinimas“* yra grįsta dinaminiu laiko skalės kraipymu ir priklauso atpažinimo, atliekant nežinomo ištarimo palyginimą su etalonais, grupei. Ši programa yra priklausoma nuo kalbėtojo. Antroji programa *„Žodžių atpažintuvas“* yra nepriklausoma nuo kalbėtojo ir atpažinimui naudoja paslėptuosius Markovo modelius. [4]

„Žodžių atpažintuvas“ yra mažo žodyno (100 žodžių) paslėptais Markovo modeliais grįsta atpažinimo sistema, kurioje naudojami nuo konteksto nepriklausomi fonemų akustiniai modeliai. [4] Jos pagrindu buvo sukurtas interneto naršyklės valdymo balsu sistemos prototipas. Atpažinimo sistemoje naudojami kalbos suvokimu, kai imituojamas žmogaus klausos aparato veikimas, grįsti požymiai – melų dažnių skalės kepstriniai koeficientai. Sistemoje galima naudoti skirtingus žodynus, visą žodį arba fonetinius vienetus reprezentuojančius modelių rinkinius. [7]

Taip pat sukurta pavienių žodžių atpažinimo ir segmentavimo sistema KAS (Kalbos Atpažinimas ir Segmentavimas). Kaip atpažinimo metodas pasirinktas dinaminio laiko skalės kraipymo metodas. Signalui analizuoti panaudotas tiesinės prognozės modelio ir tiesinės prognozės modelio kepstro analizės metodai. Apjungus segmentavimo ir atpažinimo procedūras, realizuotas pavienių žodžių atpažinimas garsais. [14]

Buvo sumodeliuota dešimties atskirai pasakytų žodžių (skaičių nuo 0 iki 9) atpažinimo sistema, naudojanti vieno paslėpto sluoksnio daugiasluoksnį perceptroną, apmokomą pagal klaidos atbulinio sklidimo algoritmą. Buvo atliktas eksperimentinis tyrimas, parodantis skirtingų požymių sistemų ir daugiasluoksnio perceptrono struktūrų naudojimo įtaką dešimties žodžių atpažinimo tikslumui. Atliekant eksperimentus sumodeliuota sistema buvo gautas didžiausias 87% žodžių atpažinimo tikslumas. [6]

Taip pat atlikti lietuvių šnekos atpažinimo tyrimai kai naudojami kitų kalbų (vokiečių, anglų, prancūzų, ispanų) atpažinimo varikliai. Tyrimai atlikti skaičių ir balso komandų atpažinimui. Aukščiausi rezultatai gauti naudojant ispanų kalbai šnekos atpažinimo variklį. [3]

3 PROJEKTINĖ DALIS

3.1 Šnekos atpažinimo įrankių ir priemonių pasirinkimas

Yra sukurta nemažai atviro kodo šnekos atpažinimo įrankių. Plačiausiai naudojami yra šie: CMU Sphinx, HTK, Julius, ISIP, Spharchcore, NICO, Intel AVSR. Iš šių minėtų įrankių labiausiai pritaikyti ištisinei šnekai atpažinti yra HTK, Julius ir CMU Sphinx.

3.1.1 Hidden Markov Toolkit (HTK) įrankis

Akustiniams, kalbos modeliams mokyti naudojamos HTK (angl. *The Hidden Markov Model Toolkit*) priemonės. HTK – programinių priemonių rinkinys, naudojamas paslėptaisiais Markovo modeliais grįsto šnekos atpažinimo tyrimams. Jį sudaro priemonės, skirtos duomenims rengti, akustiniams ir kalbos modeliams mokyti ir testuoti, akustiniams modeliams pritaikyti konkrečiam kalbėtojui, rezultatams analizuoti. Naudojantis šia programine įranga galima modeliuoti pavienių žodžių ir ištisinės šnekos atpažinimo sistemas, pasirinkti skirtingus paslėptųjų Markovo modelių parametrus ir topologijas, signalą reprezentuojančių požymių tipą. HTK priemonėms valdyti naudojama tekstinė sąsaja, o atpažinimo rezultatai pateikiami skaitine forma.

3.1.2 ATK

ATK - tai įrankis, skirtas kurti eksperimentines programas, naudojant su HTK sukurtus akustinius modelius. ATK turi C ++ sluoksnį, kuris komunikuoja su standartinėmis HTK bibliotekomis. ATK palaiko Unix ir Windows platformas.

ATK ypatybės: palaiko programavimą gijomis, audio failų įvestį/išvestį, palaiko baigtinių būsenų gramatikas ir trigramų kalbos modelius, leidžia grąžinti atpažinimo rezultatus "žodis po žodžio".

3.1.3 Julius įrankis

Julius – tai didelio našumo, atviro kodo kalbos atpažinimo sistema, skirta šnekos atpažinimo tyrėjams ir programų kūrėjams. Ji veikia paslėptųjų Markovo modelių pagrindu. Gali atlikti realaus laiko šnekos atpažinimą. Ši sistema palaiko su *HTK*, *CMU-Cam SLM* įrankiais sukurtus akustinius modelius. Su šia šnekos atpažinimo sistema galima kurti bet kokios kalbos akustinius modelius. Julius sukurtas Japonijoje ir šiuo metu vystomas ISTC konsorciumo.

3.1.4 CMU Sphinx įrankių rinkinys

CMU Sphinx yra kalbos atpažinimo sistemų rinkinys susidedantis iš:

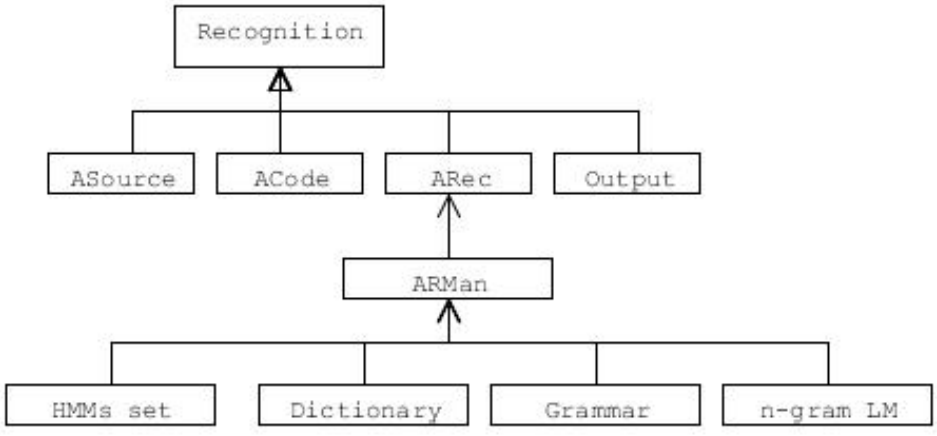
- Pocketsphinx – atpažinimo biblioteka parašyta C kalba.
- Sphinxtrain – akustinių modelių apmokymo įrankis.
- Sphinxbase – palaikymo bibliotekos reikalingos Pocketsphinx ir Sphinxtrain.
- Sphinx4 – atpažinimo biblioteka parašyta Java kalba.
- CMUcmltk – kalbos modelių įrankių rinkinio.

  Šis įrankis taip pat palaiko su HTK sukurtus akustinius modelius.

Akustinių modelių kūrimui buvo pasirinktas HTK įrankis, o stenografavimo įrankio kūrimui - ATK įrankis. Šie įrankiai pasirinkti dėl jų pritaikymo lankstumo. Su HTK įrankiu gali būti kuriami tiek pavienių žodžių, tiek ištisinei šnekai skirti akustiniai modeliai. Taip pat HTK akustiniai modeliai yra plačiai palaikomi kitų projektų.

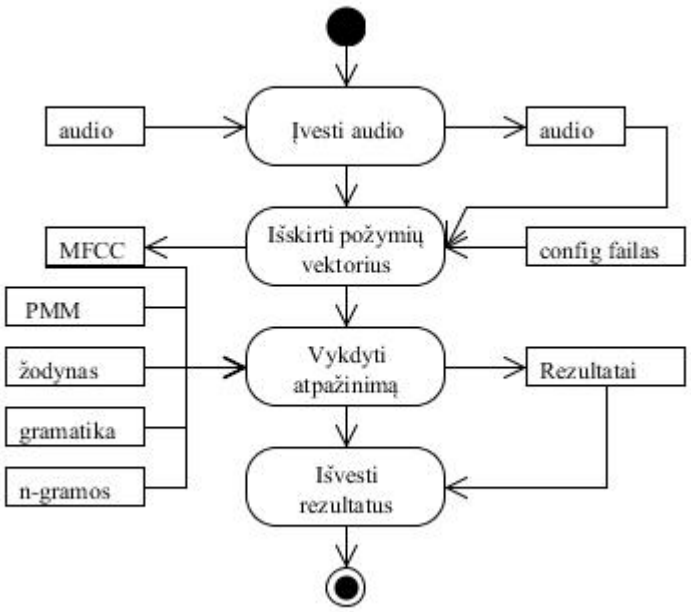
3.2 Balso įrašų stenografavimo įrankio projektas

Stenografavimo įrankis bus kuriamas, naudojant ATK bibliotekas. Programa naudos *ASource*, kuriuo bus įvedamas audio failas, *ACode* išskirs įvesto audio failo MFCC požymius, *ARec*, kuris vykdys atpažinimo etapą. Su HTK įrankiu apmokyti akustiniai modeliai, sudarytas žodynas ir gramatika bus naudojami atpažinimo etape, o gauti rezultatai išvedami į tekstinį failą (*žr. 9, 10 pav.*).



9 pav. AŠA stenografavimo įrankio koncepcinis modelis

Programos veikimas susidės iš kelių etapų: pirmiausia bus vykdoma garso failų įvestis, iš jų išskiriami MFCC požymiai, atpažinimas vykdomas panaudojant sukurtus PMM modelius, gramatiką ir žodyną. Atpažinimo rezultatai apdorojami ir išsaugomi tekstiniame faile (*žr. 10 pav.*).



10 pav. AŠA stenografavimo įrankio veiksmų diagrama

4 DARBO EIGOS APRAŠYMAS

4.1 Lietuvių kalbos automatinio šnekos atpažinimo kalbos ir akustiniai modeliai

Lietuvių kalbos automatiniam šnekos atpažinimui reikia:

- Kalbos modelio: žodyno - žodžių sąrašo, ir gramatikos - žodžių kombinacijų rinkiniai.
- Akustinio modelio, kuris statistiškai aprašytų skirtingus garsus, kuriuos sukelia kiekvienas tariamas žodis iš kalbos modelio. Norint apmokyti akustinius modelius reikia sudaryti garsyną, iš kurio bus formuojamos mokymo imtys.
- Kalbos atpažinimo variklio, kuris žmogaus ištartą garsą susietų su akustiniame modelyje esančiais garsais, o po to rastų tokią fonemų seką gramatikoje.

Lietuvoje nėra žmonių, kurie užsiimtų lietuvių kalbos garsynų kūrimu, todėl tuo turi pasirūpinti pats šnekos tyrėjas. Šiuo metu garsynus renka ir ruošia Matematikos ir Informatikos institutas, Vytauto Didžiojo, Kauno technologijos ir Vilniaus universitetai. Esantys garsynai skiriasi anotacijos lygiu, apimtimi, žodyne esančių žodžių skaičiumi ir kalbėtojų skaičiumi. [10]

Frazių garsynas. Šiam darbui skirtas garsynas buvo sudarytas iš Lietuvos radijo (LRT) mediatekoje esančių žinių laidų įrašų, kurie yra atviri visuomenei ir galimi parsisiųsti. Įrašyti šnekos signalai yra aukštos kokybės, tarimas aiškus ir teisingas. Buvo atrinkti 50 įrašų, kurie rankiniu būdu suskaidyti į sakinius ir anotuoti pagal juos atitinkančius tekstus (tekstai skelbti LRT radijo archyve (fonotekoje) kartu su atitinkamais žinių laidų įrašais) žodžių lygiu. Tekste esantys sutrumpinimai ir skaitmenys buvo perrašomi pilnais žodžiais, sintaksės žymės šalinamos. Tokiu būdu buvo sudarytas garsyne esančių žodžių sąrašas (5859 žodžiai).

Izoliuotų žodžių garsynas. Šis garsynas sudarytas iš 10 kalbėtojų (5 vyrų ir 5 moterų) 150 skirtingų žodžių, pakartotų 10 kartų, įrašų. Kiekvienas iš 150 žodžių turi 100 tarimo variantų. Kiekvienas žodis įrašytas į atskirą garso failą, kuris turi jam atitinkantį žodžio lygio anotacijos failą. Įrašai sukurti su *Audacity* programa, naudojant *Mono* kanalą, nustatant 48000 Hz dažnį ir 16 bitų formatą. Jie eksportuoti į nesuspaustą *WAV* 16 bitų *PCM* formatą. Garsyno žodžiai atrinkti pagal dažninį dabartinės rašomosios lietuvių kalbos žodyną (Grumadienė, 1997).

Su turimais paruoštais duomenimis kalbos ir akustinis modeliai bus apmokomi naudojantis *Hidden Markov Toolkit (HTK)* įrankius, kurie skirti kurti ir manipuliuoti statistinius kalbos, akustinius modelius.

4.2 Transkripcijų žodynas

Norint sukurti lietuvių šnekos akustinį modelį reikia sudaryti tarimo žodyną ir fonemų transkripcijas, įrašyti garsyną ir konvertuoti jį į formatą, tinkamą išskirti požymio vektorius. Taip pat pagal išskirtus požymio vektorius reikia apmokyti paslėptuosius Markovo modelius. Norint gauti tikslesnius rezultatus ištisinės šnekos atpažinime reikia turėti frazių garsyną su atitinkamomis tekstinėmis frazėmis ir jų transkripcijomis.

Daugelis šnekos atpažinimo sistemų yra grįstos fonemų atpažinimu. Visiems garsyne panaudotiems žodžiams reikia sudaryti transkripcijų žodyną. Norint atlikti transkribavimą reikia pasirinkti pagal kokią fonetinę sistemą bus transkribuojama. Šiame darbe fonetinė sistema sudaryta remiantis lietuvių kalbos tarties žodynu [17]. Taip sudarant transkripcijų žodyną buvo atsižvelgta į kirčiavimą, priebalsių minkštumą ir kietumą, balsių ilgumą ir trumpumą.

Pagrindiniai fonetinės transkripcijos ženklai

Transkripcijoje vartojamais ženklais žymimi:

- balsių ilgumas – pvz., [a:],
- garsų pusilgumas – pvz., [a.]
- priebalsių minkštumas – pvz., [l‘]
- [ng] – gomurinis n, vartojamas prieš [k, g];
- [ae] – paplatėjęs e;
- [e*] – balse ė žymimas garsas;
- [x] – rašyboje dviraidžiu ch žymimas garsas.
- raidėmis ą, ę,..., žymimi ilgieji garsai, pvz., [a˙]).
- kirčio ženklai: pvz., kairinis (pvz., à -> ak), dešininis (pvz.: à->ad), riestinis (pvz., ã->ar).
- dusliosios afrikatos lietuvių kalbininkų žymimos dvejopai: [c -> t’s’], [č -> t’š’].
- skardžiosios afrikatos žymimos [dz -> d’z’], [dž -> d’ž’].

Ištrauka iš transkripcijų žodyno:

```
AIŠKUS          [aiškus]        ad. i ss k u s
APSAUGA         [apsauga]       a p s a u g ak
ASMUO           [asmuo]         a s m u or
ATITIKTI        [atitikti]      a t' i t' ik k t' i
ATSKIRAS        [atskiras]      ar: t' s' k' i r a s
AUKŠTAS         [aukštas]       ad. u k ss t a s
```

4.3 Paslėptųjų Markovo Modelių apmokymas

Paslėptųjų Markovo modelių apmokymas susideda iš kelių apmokymo žingsnių. Apmokymui buvo naudojami garso įrašų failai, jau sukurti transkripcijų failai, fonemos.

Pirmiausia **HLEd** įrankiu žodžių lygio transkripcijos išplėstos į fonemų lygio transkripcijas. **HCopy** įrankiu iš visų garsyno audio failų išskiriami požymių vektoriai (*žr. 11 pav.*) ir sukuriami *MFCC* formato failai, t.y. požymių failai. Scenarijaus faile nurodoma, kokius failus reikia konvertuoti ir kur išsaugoti požymių failus. Taip pat sukuriamas konfigūracinis failas, kuriame nurodyti tokie konvertavimo parametrai:

```
SOURCEFORMAT = WAV
TARGETKIND = MFCC_0_D_A
TARGETRATE = 100000.0
SAVECOMPRESSED = T
SAVEWITHCRC = T
WINDOWSIZE = 250000.0
USEHAMMING = T
PREEMCOEF = 0.97
NUMCHANS = 26
CEPLIFTER = 22
NUMCEPS = 12
```

, čia

SOURCEFORMAT – šaltinio failo formatas *WAV*.

TARGETKIND – tikslo parametro rūšis, nurodo, kad išskiriamų požymių tipas – Melų dažnių skalės kepstro koeficientai.

TARGETRATE – tikslo modelio periodas, vienetai 100 ns.

SAVECOMPRESSED – išsaugo suspaustą išėjimo failą.

SAVEWITHCRC – prideda kontrolinę suma.

WINDOWSIZE – analizuojamo lango dydis, vienetai 250 ns.

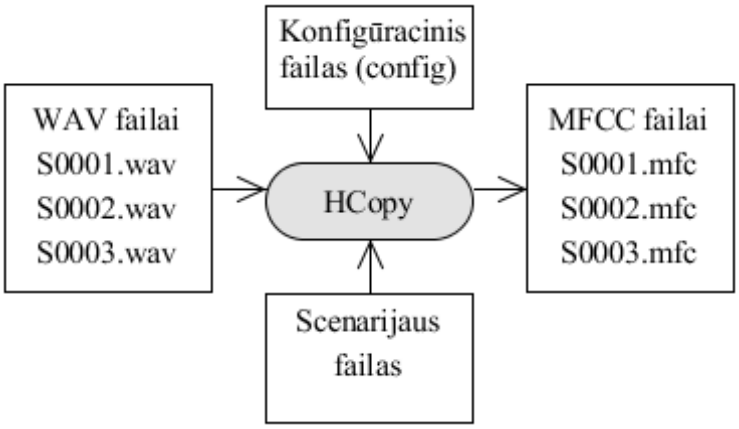
USEHAMMING – naudoja *Hamming* langą.

PREEMCOEF – nustato pradinį koeficientą.

NUMCHANS – filtro kanalų skaičius.

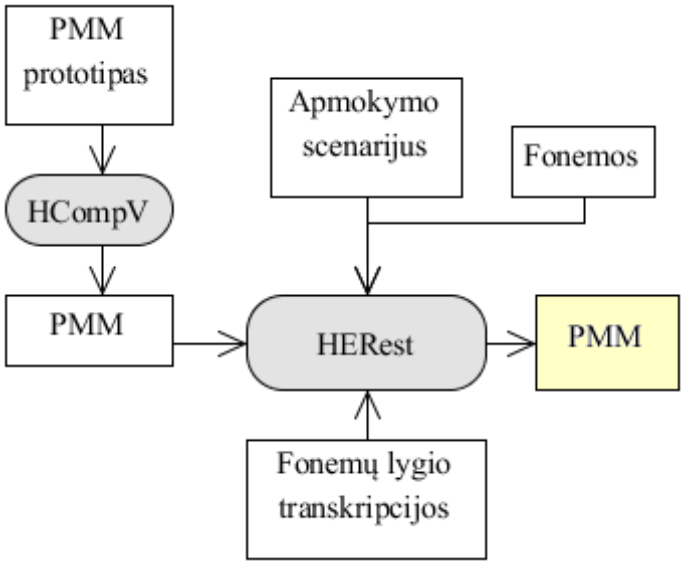
CEPLIFTER – kepstrinio kėlimo koeficientas.

NUMCEPS – kepstrinių parametrų skaičius.



11 pav. Požymių vektorių išskyrimas

PMM modelių apmokymui sukurtas prototipo modelis, aprašantis 25 *MFCC* požymių vektorius. Prototipo sukūrimo tikslas - apibrėžti modelio struktūrą. Įrankis **HCompV** peržiūri visus MFCC formato failus, suskaičiuoja vidurkį ir kovariaciją, o rezultatus priskiria duotam PMM prototipo modeliui. Prototipas išplečiamas kiekvienai fonemai. Įrankiu **HERest** fonemos iš naujo įvertinamos pagal sukurtą prototipą ir MFCC požymių vektorius (*žr. 12 pav.*).



12 pav. PMM apmokymas

Sukurti PMM modeliai buvo koreguojami - jiems pridėti tylos modelis *sp*, aprašantys trumpas pauzes, kurios gali pasirodyti ištisinėje šnekoje tarp ištariamų žodžių. Modelyje jau buvo sukurtas tylos modelis *sil*, kuris yra ilgesnis už *sp* ir naudojamas sakinio pradžioje ir pabaigoje. Tylos modelis *sp* sukuriamas **HHEd** įrankiu panaudojant *sil* tylos modelį. PMM vėl įvertinami **HERest** įrankiu jau su *sp* tylos modeliais.

Šiame etape sukurti PMM gali būti naudojami šnekos atpažinime, kuris atpažinimui naudos fonemas.

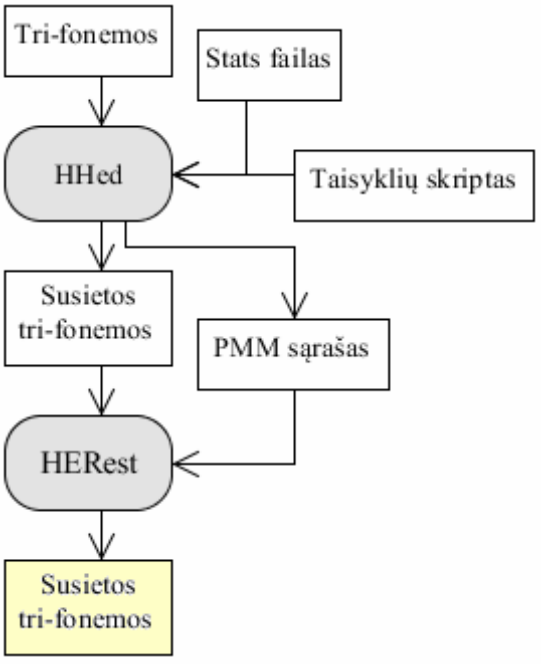
PMM apmokymas tri-fonemomis

Tri-fonemos tai trijų fonemų grupė, sudaryta pagal formą *L-X+R* (*L* - iš kairės einanti fonema, *X* - vidurinė fonema, *R* - iš dešinės einanti fonema).

Žodis sudarytas iš fonemų: EITI e i t i

Žodis sudarytas iš tri-fonemų: EITI e+i i-t+i t+i

Tri-fonemos naudojamos tikslesniam šnekos atpažinimu, kadangi tuomet yra žiūrima į fonemos kontekstą (bandoma atpažinti iš eilės einančius tris garsus, o ne vieną garsą kaip tai būtų daroma fonemų atveju). Tri-fonemos sumažina tikimybę, kad vienas garsas bus supainiotas su kitu. Tri-fonemos sudaromos konvertuojant fonemas su **HLEd** įrankiu. PMM modeliai iš naujo įvertinami įrankiu **HERest**, remiantis sukurtomis tri-fonemomis. Taip pat sukurtos skriptų failas *tree.hed*, kuriame yra taisyklės galimam fonemų klasterizavimui. **HHed** įrankiu tri-fonemos "surišamos" klasterizuojant remiantis sudarytomis taisyklėmis. PMM vėl įvertinami įrankiu **HERest** (*žr. 13 pav.*).



13 pav. PMM apmokymas tri-fonemomis

4.4 Apmokytų PMM testavimas

Šnekos atpažinimo tikslumas buvo vertinamas skaičiuojant žodžių atpažinimo procentą. Eksperimentų metu buvo tiriama, kaip kinta frazių ir pavienių žodžių atpažinimo tikslumas, kai naudojami fonemomis ir tri-fonemomis apmokyti PMM.

Frazių atpažinimo testavimui parinkti 100 sakinių iš radijo žinių įrašų, t.y. 1436 žodžiai.

Pavienių žodžių atpažinimo testavimui buvo įrašyta 740 pavienių žodžių.

Įrankiu **HBuild** sukurtas žodžių tinklas, kuris nusako, kad gali būti ištartas bet kuris žodyne esantis žodis.

Testuojama su **HVite** įrankiu. Žodžių atpažinimo tikslumui įvertinti naudojamas įrankis **HResults**.

Pirmajame tyrimo etape buvo testuojamas frazių atpažinimas. Testuojant PMM, kurie buvo apmokyti fonemomis, atpažintų žodžių procentas siekė 53,92%. Tačiau žodžių atpažinimo tikslumas siekė tik 44,63%, nes čia įvertinamas ir įterptų žodžių kiekis. Testuojant tuos pačius duomenis su tri-fonemomis apmokytais PMM tikslumas žymiai išaugo - siekė 69,35%, o atpažintų žodžių procentas - 76,67%. Iš 1873 žodžių teisingai atpažinti 1436 žodžiai.

Frazių atpažinimas fonemomis:

```
==================== HTK Results Analysis =======================
WORD: %Corr=53.92, Acc=44.63 [H=1010, D=93, S=770, I=174, N=1873]
================================================================
```

Frazių atpažinimas tri-fonemomis:

```
==================== HTK Results Analysis =======================
WORD: %Corr=76.67, Acc=69.35 [H=1436, D=59, S=378, I=137, N=1873]
================================================================
```

Žymėjimai:

WORD – žodžių atpažinimo rezultatai:

%Corr – teisingai atpažintų žodžių proc. (skaičiuojami tik teisingai atpažinti žodžiai)

Acc – žodžių atpažinimo tikslumas (atsižvelgiama ir į įterpimus)

H – teisingai atpažintų žodžių skaičius

D - praleistų (ištrintų) žodžių skaičius

S – sukeistų žodžių skaičius

I – įterptų žodžių skaičius

N –atpažinimui pateiktų žodžių skaičius

Antrajame tyrimo etape buvo testuojamas pavienių žodžių atpažinimas. Testuojant PMM, kurie buvo apmokyti fonemomis, atpažintų žodžių procentas siekė 29,32%. Iš 740 žodžių teisingai atpažinti tik 217 žodžiai, žodžių atpažinimo tikslumas siekė tik 15,68%. Testuojant tuos pačius duomenis su tri-fonemomis apmokytais PMM tikslumas žymiai išaugo - siekė 61,76%, o atpažintų žodžių procentas - 75,14%. Iš 740 žodžių teisingai atpažinti 556 žodžiai.

Pavienių žodžių atpažinimas fonemomis:

```
================ HTK Results Analysis =========================
WORD: %Corr=29.32, Acc=-15.68 [H=217, D=8, S=515, I=333, N=740]
===============================================================
```

Pavienių žodžių atpažinimas tri-fonemomis:

```
================ HTK Results Analysis =========================
WORD: %Corr=75.14, Acc=61.76 [H=556, D=0, S=184, I=99, N=740]
===============================================================
```

Atlikti pirmieji du tyrimo etapai parodė, kad naudojant tri-fonemas žodžių atpažinimo tikslumas žymiai išauga. PMM apmokyti tri-fonemomis duoda geresnius atpažinimo rezultatus nei fonemomis apmokyti PMM tiek pavienių žodžių atpažinime, tiek frazių atpažinime. Toliau tyrimai buvo atliekami tik su tri-fonemomis.

Trečiasis tyrimo etapas atliktas, norint patikrinti, ar fonemų, skirtų žodžių transkripcijoms, kiekio sumažinimas turės įtakos atpažinimo tikslumui. Buvo sudarytos naujos žodžių fonetinės transkripcijos, kuriose atsisakyta garsų pusilgumą, priebalsių minkštumą ir kirčio ženklus žyminčios fonemos - jos priskirtos atitinkamoms fonemoms (balsėms ir priebalsėms).

Buvo testuojama su frazių atpažinimui skirta įrašų aibe. Eksperimento rezultatai parodė, kad didelės įtakos atpažinimo tikslumui tai neturėjo - sumažinus fonemų kiekį žodžių atpažinimo tikslumas padidėjo **2%**, kadangi sumažėjo įterptų žodžių kiekis (*I=51*).

```
====================== HTK Results Analysis ======================
WORD: %Corr=75.33, Acc=72.61 [H=1411, D=69, S=393, I=51, N=1873]
==================================================================
```

Ketvirtasis tyrimo etapas atliktas žinant, kad kuriamas stenografavimo įrankis bus skirtas ištisinės šnekos atpažinimui, todėl reikėjo padidinti frazių atpažinimo tikslumą. Tai būtų galima padaryti sukurtą gramatiką (žodžių tinklą) praplečiant bigramomis.

Bigramos apmokytos iš žinių laidų įrašus (kuriais buvo apmokyti PMM modeliai) atitinkančių sakinių. Tam panaudotas HTK įrankis **HLStats**. Buvo testuojama su frazių atpažinimui skirta įrašų aibe. Žodžių atpažinimo tikslumas žymiai padidėjo ir siekė 94%.

```
====================== HTK Results Analysis =======================
WORD: %Corr=96.32, Acc=94.98 [H=1804, D=19, S=50, I=25, N=1873]
====================================================================
```

Tačiau pavienių žodžių atpažinimo tikslumui bigramos įtakos neturėjo - jis išliko toks pat.

Integravus šią gramatiką į stenografavimo įrankį jis taps nuo konteksto priklausomu, t.y. šiuo atveju tai bus žinių įrašams skirtas stenografavimo įrankis.

Išanalizavus testavimo rezultatus buvo pastebėta, kad dažniausiai neatpažįstami yra panašiai skambantys žodžiai, tokie kaip: lėšos - šios, įstatymui - įstatymai, projektą - projekte, kalba - kalbą, su dar - sudarė ir t.t. Taip pat daug praleistų trumpų žodžių, tokių kaip: į, ir, o, iš. Todėl reikėtų didesnio kiekio duomenų, kuriais būtų galima apmokyti prastai atpažįstamų žodžių akustinius modelius.

Praplėtus žodyną ir bigramas apmokius didesniu kiekiu sakinių, būtų galima pagerinti pavienių žodžių ir trumpų frazių, sudarytų iš dviejų trijų žodžių, atpažinimą.

Ištisinės šnekos atpažinimo atveju, žodžių tinklo praplėtimui būtų galima panaudoti trigramas. Trigramų apmokymui ir jų testavimui būtų galima panaudoti **HDecode** įrankį, kuris yra suderinamas su HTK.

4.5 Stenografavimo įrankio prototipo kūrimas

Stenografavimo įrankis kurtas naudojant ATK biblioteką, kuri yra suderinama su HTK įrankiu sukurtais akustiniais modeliais. Vartotojas programą iškviečia iš komandinės eilutės. Iškviečiant programą nurodomi parametrai: konfigūracinio failo, kuris aprašo įrankio šnekos atpažinimo parametrus ir tekstinio failo pavadinimas, kuriame nurodyti audio failai stenografavimui (*žr. 14 pav.*).

14 pav. Stenografavimo programos panaudos atvejų diagrama

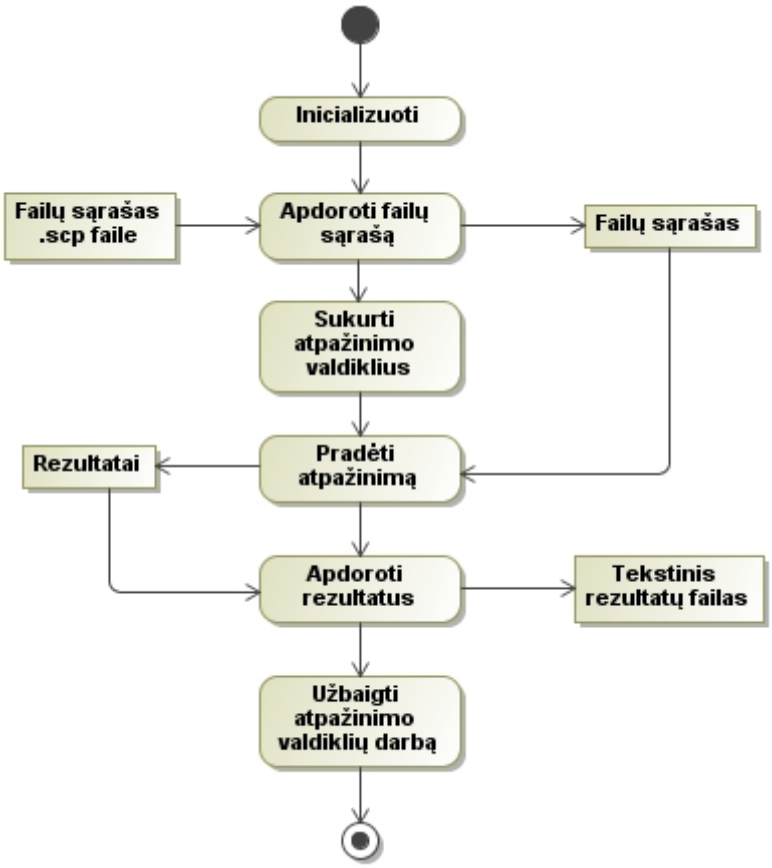
Stenografavimo įrankio koncepcinis modelis pasikeitė *(žr. 15 pav.)*. Realizuotą stenografavimo įrankį sudaro tokios pagrindinės dalys:

- inicializavimą vykdanti paprogramė,
- failų sąrašą apdorojanti paprogramė,
- atpažinimą vykdanti paprogramė,
- atpažinimo rezultatus apdorojanti paprogramė,
- už vartotojo sąsają ir kitų programos dalių veikimą atsakinga pagrindinė programa.

15 pav. Stenografavimo įrankio koncepcinis modelis

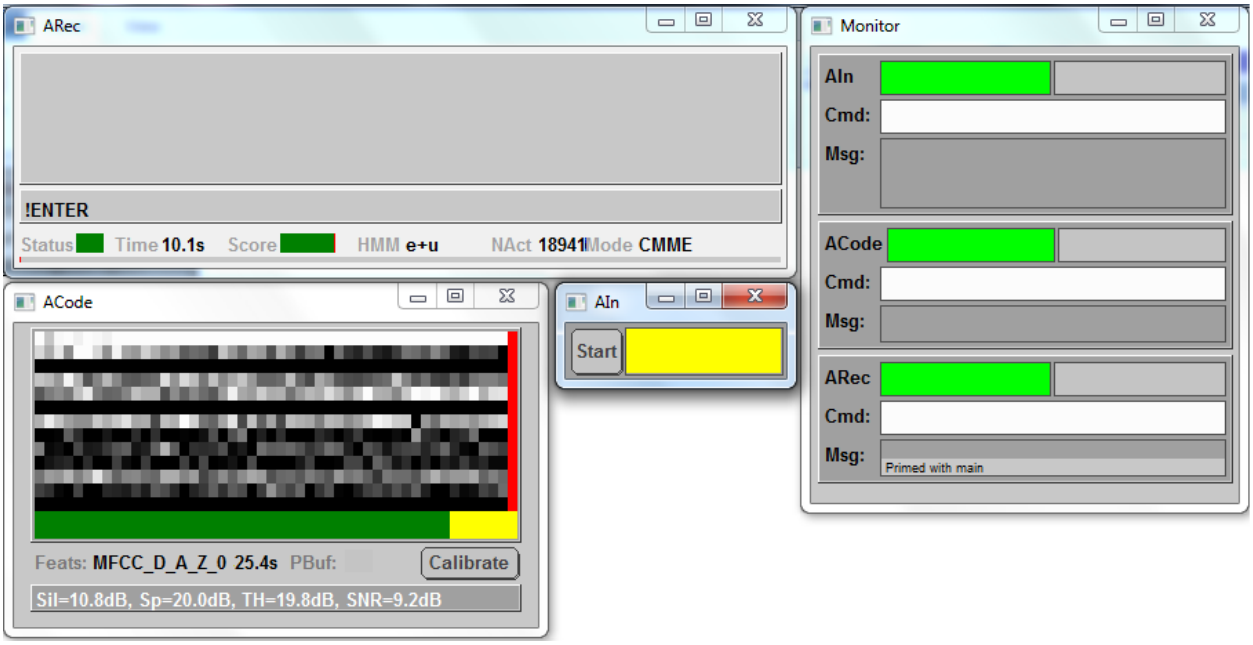
Programos veikimas taip pat šiek tiek pakito negu buvo numatyta pradiniame programos projekte (*žr. 16 pav.*). Jis prasideda patikrinant ar programos paleidimo metu nurodyti konfigūracinis ir garso įrašų sąrašo failai. Sukuriant atpažinimo valdiklius yra inicializuojami

resursų valdiklis *ARMan*, kuris atsakingas už PMM modelių, gramatikos, žodyno integravimą į programą. Duomenys apie PMM modelius, gramatiką ir žodyną gaunami iš konfigūracinio failo, nurodyto paleidžiant programą.



16 pav. Stenografavimo įrankio veiksmų diagrama

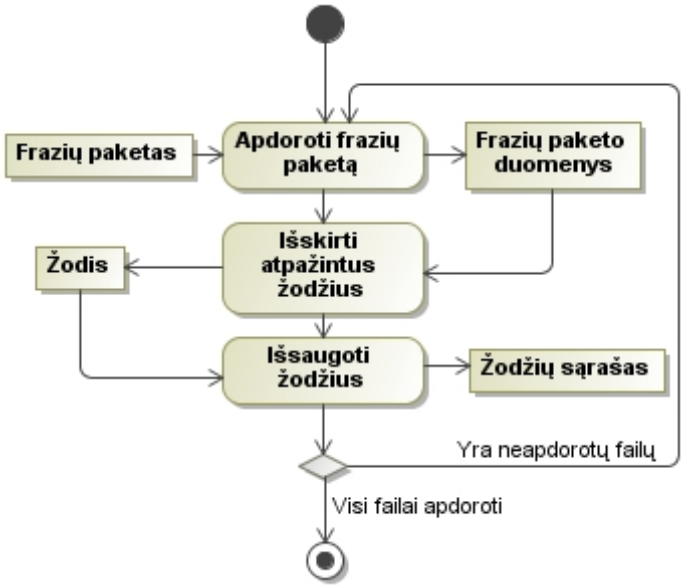
Kitame etape yra sukuriami vartotojo sąsajos komponentai *AMonitor*, skirti stebėti atpažinimo eigą (*žr. 17 pav.*). Į vartotojo sąsajos komponentus įtraukiami: *ACode* (kuris yra atsakingas už požymių vektorių išskyrimo eigą), *ARec* (kuris yra atsakingas už šnekos atpažinimo eigą), *ASource* (kuris yra atsakingas už audio signalo įvesties eigą). Atpažinimas vykdomas tik tada, jei yra nurodyti stenografavimui skirti garso failai (audio failai turi būti *.wav* formato). Pradėjus atpažinimą pagal konfigūraciniame faile nurodytus MFCC parametrus pradedami apdoroti audio failai.



17 pav. Vartotojo sąsajos komponentai

Pagal integruotą PMM modelį apdorojant failus renkami *APacket* tipo duomenys apie šnekos atpažinimo paketus ir išskiriami *APhraseData* tipo atpažinimo rezultatus saugantys paketai (*žr. 18 pav.*). Išskirti duomenys yra konvertuojami į *String* tipo eilutę ir po to perduodami rezultatus apdorojančiai paprogramei, kuri juos įrašo į tekstinį failą.

Pabaigus audio failų atpažinimą ir rezultatų išvedimą, visų atpažinimo valdiklių darbas yra sustabdomas.



18 pav. Atpažinimo rezultatų gavimas

Programos veikimas ištestuotas su ATK bibliotekų rinkinyje esančiais testiniais duomenimis: žodynu, PMM modeliais, gramatika, audio failais. Programos veikimas yra korektiškas - atpažintus žodžius įrašo į tekstinį failą. Integravus HTK įrankiu sukurtus PMM modelius iškilo problema, kai neteisingai atpažįstami audio įrašo žodžiai. Taip yra dėl konfigūraciniame faile nurodytų audio signalų apdorojimo ir integruojamų PMM modelių parametrų. Nors pagal ATK dokumentaciją HTK ir ATK akustinių modelių rinkiniai yra suderinami, parametrų rinkiniai tinkantys PMM modeliams, sukurtais su HTK, nėra tinkami ATK sistemoje. Pavyzdžiui, programą testuojant su HVite įrankiui skirtais audio signalo apdorojimo parametrais, žodžių seka "aiškus balsas" atpažįstama kaip "darbas", nors su HVite įrankiu ji atpažįstama teisingai.

Panaudoti audio signalo apdorojimo parametrai:

```
SOURCEFORMAT      = WAV
SOURCERATE        = 625
SOURCEKIND        = WAVEFORM
TARGETKIND        = MFCC_0_D_A
TARGETRATE        = 100000.0
WINDOWSIZE        = 250000.0
ENORMALISE        = F
ZMEANSOURCE       = F
USEHAMMING        = T
PREEMCOEF         = 0.97
USEPOWER          = T
NUMCHANS          = 26
CEPLIFTER         = 22
NUMCEPS           = 12
SILFLOOR          = 50.0
```

4.6 Darbo rezultatų analizė

Sukurtas balso įrašų stenografavimo įrankio prototipas, atpažinimo procesą atliekantis iš audio failų ir rezultatus išsaugantis tekstiniame faile. Įrankyje integruoti PMM metodu apmokyti akustiniai modeliai, ištisinei šnekai nepriklausomai nuo kalbėtojo atpažinti. Sukurta gramatika, praplėsta bigramomis, apmokytomis žinių laidų tekstais, ir sudarytas žodynas, kuriame yra virš 5500 skirtingų žodžių.

Atsižvelgiant į gautus PMM testavimo rezultatus, šiuo metu negalime sakyti, kad dabartinis įrankis yra pritaikomas efektyviam naudojimui. Tam yra kelios priežastys. Pirmoji priežastis kyla iš to, kad PMM apmokymui reikėtų didesnio duomenų kiekio, kad pagal akustinius modelius būtų galima atpažinti univesalius įrašus, nes dabar ji orientuota tik į žinių laidų įrašus. Iš to seka antroji - žodyno dydžio problema, nes didinant apmokymo duomenų dydį reikia didinti ir žodyne esančių žodžių kiekį. Gramatika šiuo atveju būtų tikslinga praplėsti trigramomis. Kadangi sistema geriau atpažįsta rišlias frazes, reikėtų didinti pavienių žodžių atpažinimo tikslumą.

4.7 Rekomendacijos

Pavienių žodžių atpažinimui pagerinti būtų galima praplėsti žodyną ir bigramas apmokyti didesniu kiekiu sakinių. Ištisinės šnekos atpažinimui būtų galima trigramomis praplėsti žodžių tinklą.

Šnekos atpažinimo tikslumo padidinimui apmokytus PMM akustinius modelius būtų galima iš naujo įvertinti HHEd įrankiu padidinus PMM būsenų mišinių skaičių. Tam tinkamiausia tokia eiga: didinti PMM būsenų mišinių skaičių 2-3 kartus ir įvertinti naujus modelius HERest įrankiu, ir vėl padidinti PMM būsenų mišinių skaičių 2-3 kartus. Taip bus galima stebėti, koks mišinių kiekis duoda geriausius rezultatus.

5 APIBENDRINIMAS IR IŠVADOS

- Kuriamas garso įrašų stenografavimo įrankio prototipas priklausytų tokioms AŠA klasifikacijoms: atpažįsta ištisinę šneką, yra nepriklausoma nuo kalbėtojo, turi vidutinį žodyną.
- Išnagrinėjus šnekos signalo požymių išskyrimo metodus atpažinimui buvo pasirinktas MFCC, nes šis metodas parodė geriausius rezultatus lietuvių kalbos šnekos atpažinime.
- Išnagrinėjus metodus, skirtus automatiniam šnekos atpažinimui, tolimesniam darbui pasirinkti paslėptieji Markovo modeliai kadangi jie gali būti naudojami ištisinės šnekos atpažinimui.
- Sukurti izoliuotų žodžių ir frazių garsynai, kadangi ištisinės šnekos atpažinime tikslesniems rezultatams gauti reikalingi kuo daugiau nuo šnekėjimo tipo priklausomų duomenų. Taip pat remiantis lietuvių kalbos tarties žodynu sukurti žodyno ir transkripcijų failai.
- Tyrimų rezultatai parodė, kad PMM modeliai apmokyti tri-fonemomis duoda geresnius atpažinimo rezultatus nei fonemomis apmokyti PMM. Geresni atpažinimo rezultatai gauti tiek pavienių žodžių atpažinime (61%), tiek frazių atpažinime (69%). Todėl rekomenduojama šnekos atpažinimui naudoti tri-fonemas.
- Atlikus fonemų kiekio transkripcijose įtakos atpažinimo tikslumui tyrimus, kai buvo atsisakyta kirčio ženklus, garsų pusilgumo ir minkštumo žyminčių fonemų, paaiškėjo, kad tai didelės įtakos atpažinimo tikslumui neturėjo - atpažinimo tikslumas padidėjo 2%.
- Atlikus tyrimus su bigramomis praplėstu žodžių tinklu buvo nustatyta, kad bigramos pagerina ištisinės šnekos atpažinimo tikslumą. Žodžių atpažinimo tikslumas siekė 94%, tačiau pavienių žodžių atpažinimui tai įtakos neturėjo, todėl reikėtų duomenų, kuriais būtų galima apmokyti prastai atpažįstamų žodžių akustinius modelius.

6 LITERATŪROS SĄRAŠAS

1. ANUSUYA, M. A.; KATTI, S. K. . *Speech Recognition by Machine: A Review*. In *(IJCSIS) International Journal of Computer Science and Information Security* [interaktyvus] 2009. [žiūrėta 2013-11-01] Prieiga per internetą: <http://arxiv.org/ftp/arxiv/papers/1001/1001.2267.pdf>.
2. *Automatic Speech Recognition* [interaktyvus]. 2006 [žiūrėta 2013-10-25]. Prieiga per internetą: <http://isoft.postech.ac.kr/~gblee/Course/CS703AHLT/postech-ASR.ppt>.
3. BARTISIUTE, G.; RATKEVICIUS, K., *Speech Server based Lithuanian Voice Commands Recognition.* In *Elektronika ir elektrotechnika* [interaktyvus]. 2012 [žiūrėta 2013-11-11]. Prieiga per internetą: <http://www.eejournal.ktu.lt/index.php/elt/article/view/3061/2125>.
4. BARTOŠEVIČ, Daniel; KAMARAUSKAS, Juozas. *Nepriklausomas nuo kalbėtojo balso komandų atpažinimo tyrimas* [interaktyvus]. 2011 [žiūrėta 2013-11-11]. Prieiga per internetą: <dspace.vgtu.lt/bitstream/1/737/1/DBartosevic.pdf>.
5. BIVAINIS, Robertas. *Balso atpažinimo programų lietuvinimo galimybių tyrimas* [interaktyvus]. 2013 [žiūrėta 2013-11-11]. Prieiga per internetą: <http://vddb.library.lt/fedora/get/LT-eLABa-0001:E.02~2013~D_20130930_090749-15587/DS.005.0.01.ETD >.
6. FILIPOVIČ, Mark. *Atskirai pasakytų žodžių atpažinimo, naudojant neuroninius tinklus, tyrimas* [interaktyvus]. [žiūrėta 2013-11-11]. Prieiga per internetą: <http://www.ktu.lt/lt/apie_renginius/konferencijos/2006/k6_02/IT2003/Sekcija09.pdf >.
7. FILIPOVIČ, Mark; RINGELIENĖ, Živilė. *Žodžių atpažinimo, grįsto paslėptaisiais Markovo modeliais, vizualizavimo ir analizės programinė įranga.* In *Informacinės technologijos ir kalba* [interaktyvus]. 2011 [žiūrėta 2013-11-10] Prieiga per internetą: <http://www.leidykla.eu/fileadmin/Informacijos_mokslai/2011-56/63-72.pdf>.
8. GAIKWAD, S. K.; GAWALI, B. W.; YANNAWAR, Pravin. A Review on Speech Recognition Technique. In *International Journal of Computer Applications (0975 – 8887)* [interaktyvus] 2010 [žiūrėta 2013-11-15]. Prieiga per internetą: < http://www.ijcaonline.com/volume10/number3/pxc3871976.pdf>.
9. KEMBLE, Kimberlee. *An Introduction to Speech Recognition* [interaktyvus]. [žiūrėta 2013-10-18]. Prieiga per internetą:

<ftp://service.boulder.ibm.com/software/partners/comarketing/na/ss/we/WS_Voice_Server_White_Paper.pdf>.

10. LAURINČIUKAITĖ, Sigita. Lietuvių šnekos atpažinimo akustinis modeliavimas. Daktaro disertacija [interaktyvus]. 2008 [žiūrėta 2013-10-18]. Prieiga per internetą: < http://www.mii.lt/files/mii_dis_08_laurinciukaite.pdf>.
11. LILEIKYTĖ, Rasa. *Šnekos atpažinimo požymių kokybės vertinimas.* Daktaro disertacija [interaktyvus]. 2012 [žiūrėta 2013-11-12]. Prieiga per internetą: <http://www.mii.lt/files/mii_dis_2012_lileikyte.pdf >.
12. SINHA, Priyabrata. *Speech Processing in Embedded Systems* [interaktyvus]. [žiūrėta 2013-11-14]. Prieiga per internetą: <bok%3A978-0-387-75581-6.pdf>.
13. SLEDEVIČ, Tomyslav; STAŠIONIS, Liudas. *Lietuvių kalbos Pavienių žodžių atpažinimo algoritmo įgyvendinimas LAUKU Programuojama Logine Matrica.* In *Elektronika ir elektrotechnika* [interaktyvus]. 2013 [žiūrėta 2013-11-14]. Prieiga per internetą: <www.mla.vgtu.lt/index.php/mla/article/download/mla.2013.18/pdf>.
14. TAMULEVIČIUS, G. *Pavienių žodžių atpažinimo sistemų kūrimas*. Daktaro disertacija [interaktyvus]. 2008 [žiūrėta 2013-10-28]. Prieiga per internetą: <http://donelaitis.vdu.lt/disertacijos/Tamulevicius_2008.pdf>.
15. VAIČIŪNAS, Airenas. *Lietuvių kalbos statistinių modelių ir jų taikymo šnekos atpažinimui tyrimas, kai naudojami labai dideli žodynai*. Daktaro disertacija [interaktyvus]. 2006 [žiūrėta 2013-11-04]. Prieiga per internetą: <http://donelaitis.vdu.lt/disertacijos/Vaiciunas_2006.pdf >.
16. VIMALA, C; RADHA, V. *A Review on Speech Recognition Challenges and Approaches.* In *World of Computer Science and Information Technology Journal (WCSIT)* [interaktyvus]. 2012 [žiūrėta 2013-11-11]. Prieiga per internetą: <http://www.wcsit.org/pub/2012/vol.2.no.1/A%20Review%20on%20Speech%20Recognition%20Challenges%20and%20Approaches.pdf>.
17. KAZLAUSKIENĖ, Asta; RAŠKINIS, Gailius. Lietuvių kalbos tarties žodynas [elektroninis išteklius]. Kaunas: Vytauto Didžiojo universitetas, 2012. ISBN 9789955128571.

7 ANOTACIJA (SUMMARY)

Autorius: Jūratė Vaičiulytė

Tema: Automatinio šnekos atpažinimo metodų tyrimas ir taikymai balso įrašams stenografuoti

Šiaulių universitetas 2015.

Šiame darbe išanalizuoti metodai, skirti automatiniam šnekos atpažinimui. Paslėptųjų Markovo Modelių metodu apmokyti akustiniai modeliai, sukurtas kalbos modelis ir žodynas. Sukurti modeliai integruoti į stenografavimo įrankio prototipą, kuris skirtas balso įrašų stenografavimui. Remiantis akustinių modelių testavimo rezultatais pasiektas 94% frazių atpažinimo tikslumas.

Author: Jūratė Vaičiulytė

Subject: The Investigation of Automatic Speech Recognition Methods and Their Applications in the Stenography of Voice Records

Šiauliai University 2015.

Analysis of automatic speech recognition approaches was performed. In the flow of this work, Hidden Markov Models (acoustic models), language model and dictionary were created. These models was integrated into voice records stenography application. Based on testing results of acoustic models an accuracy of 94% was achieved.

8 PRIEDAI

1 priedas

CD turinys

Kataloguose:

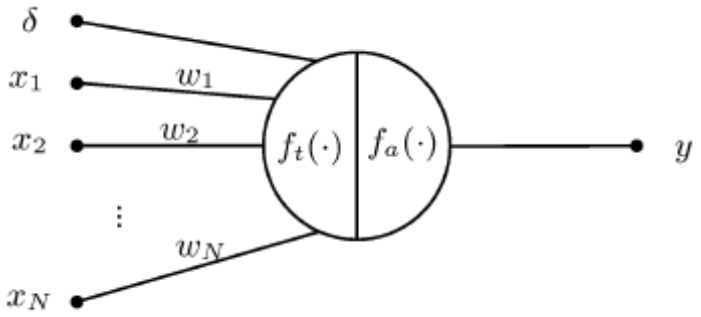
- ASten - audio įrašų stenografavimo įrankis
    - \programos naudojimosi instrukcija.txt
- ATK įrankis - ATK bibliotekų rinkinys
- PMM modeliai - sukurti PMM modeliai, jų testavimo rezultatų failai
    - \irasai\ - 10-ties kalbėtojų įrašais apmokyti PMM modeliai
    - \zinios\ - PMM modeliai žinių įrašams
    - \zinios2\ - PMM modeliai žinių įrašams apmokyti su pakeistomis žodžių transkripcijomis
    - \zinios3\ - PMM modeliai žinių įrašams (PMM su padidintu būsenų mišinių kiekiu)
    - \komandų naudojimosi instrukcija.txt
- HTK įrankis - HTK įrankių rinkinio instaliaciniai failai
- J.Vaiciulyte_Automatinis_snekos_atpazinimas - darbo aprašas .doc ir .pdf formatais.

2 priedas

Neuroniniais tinklais grįsti metodai

Užuot naudojami bendrosios paskirties šnekos atpažinime DNT gali apdoroti prastos kokybės, triukšmingus duomenis ir sistemų nepriklausomybę nuo kalbėtojo. Tokios sistemos gali pasiekti didesnį tikslumą, nei PMM grįstos sistemos, tačiau tam reikia apmokymo duomenų ir riboto žodyno. Tačiau reikėtų pasakyti, jog neuronų tinklai nepasiteisino kaip savarankiškas atpažinimo metodas (ypač ištisinės kalbos atpažinime) ir dažnai naudojami kartu su paslėptaisiais Markovo modeliais. Tokios hibridinės sistemos dažniausiai naudoja neuroninio tinklo metodus fonemų atpažinimo daliai ir PMM kalbos modeliavimo daliai. [5,16]

Pagrindinis visų dirbtinių neuronų tinklų elementas – neuronas – supaprastintas biologinio neurono modelis, sudarytas iš branduolio su įėjimo ir išėjimo taškais (*žr. 19 pav*). [14]



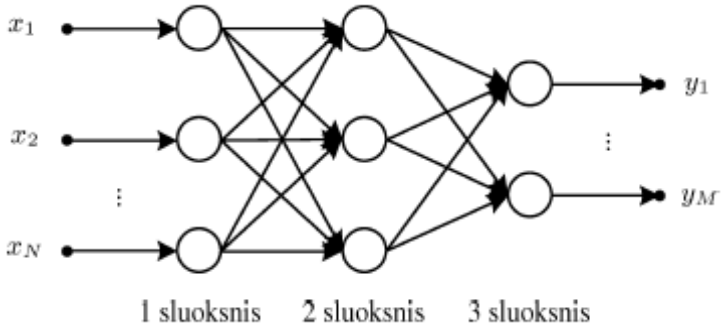
19 pav. Neurono modelis [14]

Neurono branduolį sudaro dvi funkcijos: tinklo ir aktyvuojanti. Tinklo funkcija nulemia įėjimo duomenų apdorojimą. Aktyvuojanti funkcija formuoja neurono išėjimo signalą, tiesiškai arba netiesiškai transformuodama įėjimų signalų darinį. Tokio neurono modelio išėjimo signalas apibrėžiamas

$$y = f_a(\sum_{i=1}^{N} w_i x_i - \delta) \qquad (8)$$

čia x_i – i-asis neurono įėjimas, w_i – įėjimo svorio koeficientas, δ – nuolatinė dedamoji, veikianti kaip įėjimo slenkstis, f_a – aktyvuojanti funkcija, y – neurono išėjimas. [14]

Jungdami atskirus neuronus į masyvus gauname taip vadinamus neuronų tinklus. Tinklai apibūdinami jų topologija, t.y. neuronų sujungimo schema. Daugiasluoksniame perceptrone neuronai yra organizuoti tarpusavyje sujungtomis grupėmis, taip vadinamais sluoksniais (*žr. 20 pav*.). Sluoksniai, kurių išėjimo tiesiogiai nestebime, vadinami paslėptais. [14]



20 pav. Daugiasluoksnio perceptrono su dviem paslėptais sluoksniais struktūra [14]

Viena svarbiausių neuronų tinklų savybių – sugebėjimas mokytis, t.y. keisti struktūrą pagal gaunamus duomenis. Mokymo procesas vyksta koreguojant svorio koeficientus w_i. Mokymas gali būti su mokytoju arba be mokytojo. Mokymo su mokytoju (taikomi perceptronui) metu tinklui pateikiama mokymo aibė, sudaryta iš įėjimo duomenų ir laukiamo išėjimo, o korekcija vykdoma pagal gaunamų rezultatų atitikimą pateiktiesiems. Mokymui be mokytojo (naudojamam Kohoneno tinklams) pateikiami tik įėjimo duomenys, o svorio koeficientų korekcija vyksta siekiant panašiems įėjimo signalams gauti atitinkamo panašumo išėjimo signalus. [14]

Neuronų tinklų sugebėjimas atvaizduoti N įėjimų į M išėjimų daro juos tinkamais klasifikacijai. Kaip įėjimo signalus pateikę šnekos pavyzdžio analizės duomenis (požymių vektorius ar jų elementus) galime atlikti šnekos pavyzdžių klasifikaciją. Tinklams mokinti tokiu atveju tenka naudoti segmentuotą ir žymėtą kalbos signalą. Segmentavimo lygmenį lemia nagrinėjamos šnekos pavyzdys – atpažįstant fonemas signale turi būti nurodytos fonemų ribos ir jas atitinkančios klasės žymė, pavienių žodžių atveju – žodžių ribos. [14]

Kalbai atpažinti dažniausiai naudotas daugiasluoksnis perceptronas. Nedidelių žodynų atveju atpažinimas neuronų tinklais tikslumu beveik prilygo paslėptiems Markovo modeliams. Neigiama perceprtono savybė – pakankamai sudėtingas apmokymo etapas. Tinklui mokinti reikia turėti segmentuotus ir žymėtus šnekos pavyzdžius, o pats mokymas yra pakankamai ilgas procesas. [14]

Šnekos požymių išskyrimo metodai

Tiesinio prognozavimo metodas (*angl. Linear Predictive Coding (LPC)*)

Vienas iš efektyviausių skaitmeninių signalų analizės metodų yra tiesinio prognozavimo metodas (*žr. 21 pav*.). Pagrindinė idėja yra ta, kad LPC šnekos signalo pavyzdys gali būti aproksimuojamas kaip tiesinė pastarųjų šnekos signalų pavyzdžių kombinacija. Minimizuojant įėjimo šnekos signalo ir prognozuojamų reikšmių kvadratų skirtumų sumą (iš baigtinio intervalo), gali būti nustatytas unikalus parametrų arba prognozavimo koeficientų rinkinys. Šie koeficientai sudaro LPC šnekos signalo pagrindą. LPC analizė leidžia apskaičiuoti šnekos signalo tiesinį prognozavimo modelį laike. Prognozuoti koeficientai yra transformuojami į parametrus, vadinamus Kepstro koeficientais (angl. *Cepstral coefficients*). [16]

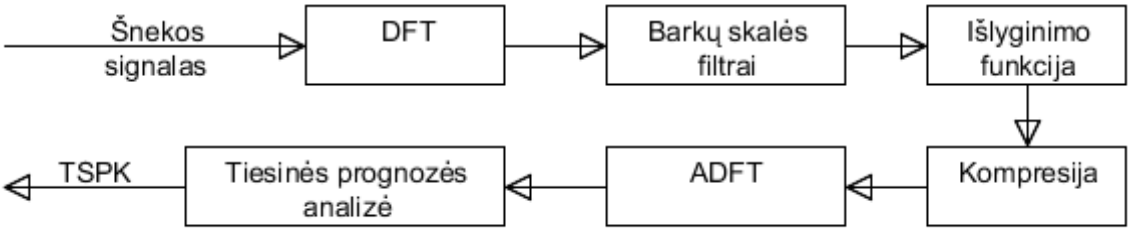
21 pav. Signalo požymių išskyrimas tiesiniu prognozavimo metodu [16]

Tiesinės suvokimo prognozės modelio analizės požymiai

Tiesinė suvokimo prognozė (angl. *Perceptual Linear Prediction*) (*žr. 22 pav*.) modeliuoja klausos suvokimu grįstą spektrą. Tam naudojama visų polių funkcija bei tiesinės prognozės autokoreliacijos technika. Tiesinis suvokimo prognozės modelis skiriasi nuo tiesinio prognozės modelio, nes šiuo atveju modelis naudoja: [11]

- barkų skalės dažnių filtrus;
- garsumo išlyginimą (angl. *equal–loudness preemphasis*);
- kubinės šaknies garsumo intensyvumo kompresiją (angl. *cubic–root intensity loudness compression*).

Šie papildomi apdorojimo etapai modeliuoja žmogaus klausos sistemos savybes. [11]



22 pav. Signalo požymių išskyrimas tiesinio suvokimo prognozės metodu [11]