МІНІСТЭРСТВА АДУКАЦЫІ РЭСПУБЛІКІ БЕЛАРУСЬ

БЕЛАРУСКІ ДЗЯРЖАЎНЫ ЎНІВЕРСІТЭТ

ФАКУЛЬТЭТ ПРЫКЛАДНОЙ МАТЭМАТЫКІ І ІНФАРМАТЫКІ

Кафедра дыскрэтнай матэматыкі і алгарытмікі

ЗАХАРАВА Вікторыя Валер’еўна

АЛГАРЫТМЫ ТЭКСТАНЕЗАЛЕЖНАЙ ІДЭНТЫФІКАЦЫІ ДЫКТАРАЎ ДЛЯ АНГЕЛЬСКАЙ І БЕЛАРУСКАЙ МОЎ

Магістэрская дысертацыя

1-31 81 09 «Алгарытмы і сістэмы апрацоўкі вялікіх аб’емаў дадзеных»

Навуковы кіраўнік

Гецэвіч Юрый Станіслававіч,

кандыдат тэхнічных навук,

заг. лабараторыі распазнавання і сінтэза

маўлення АІПІ НАН Беларусі

Дапушчана да абароны

«___» _____ 2018 г.

Заг. кафедры дыскрэтнай матэматыкі і алгарытмікі

_____ У.М. Котаў

доктар фізіка-матэматычных навук, прафесар

Мінск, 2018

Змест

АГУЛЬНАЯ ХАРАКТАРЫСТЫКА ПРАЦЫ

Магістэрская дысертацыя, 64 с., 38 мал., 3 табл., 28 крыніц, 3 дадатка.

ТЭКСТАНЕЗАЛЕЖНАЯ ІДЭНТЫФІКАЦЫЯ ДЫКТАРА, БІЯМЕТРЫЯ, МАДЭЛІ ГАУССАВЫХ СУМЕСЯЎ, ВЫБІРАННЕ ПРЫМЕТ, КЕПСТРАЛЬНЫЯ КАЭФІЦЫЕНТЫ, ФАНЕМНАЯ СЕГМЕНТАЦЫЯ

Аб'ект даследавання - лічбавыя запісы маўленчых сігналаў.

Прадмет даследавання - тэкстанезалежная аўтаматычная сістэма ідэнтыфікацыі чалавека па голасу.

Мэта працы - распрацоўка сістэмы тэкстанезалежнай ідэнтыфікацыі дыктара на аснове мадэлей гауссавых сумесяў і тэставанне яе працы на датасэтах англійскай і беларускай моў.

Метады даследавання - аналіз існуючых алгарытмаў, вывучэнне аналагаў, спектральны аналіз, аўдыёметад, мадэляванне, эксперымент, параўнанне, вывучэнне дакументацыі бібліятэк распазнання мовы.

Вынікам працы з'яўляецца праграма, якая рэалізуе кампаненты сістэмы тэкстанезалежнай ідэнтыфікацыі дыктара на аснове фанемнай сегментацыі і мадэлей гауссавых сумесяў.

Вобласці прымянення - кантакт-цэнтры, крыміналістычная экспертыза, сістэмы электроннага гандлю, ахоўныя сістэмы.

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Магистерская диссертация, 64 с., 38 рис., 3 табл., 28 источников, 3 приложения.

ТЕКСТОНЕЗАВИСИМАЯ ИДЕНТИФИКАЦИЯ ДИКТОРА, БИОМЕТРИЯ, МОДЕЛИ ГАУССОВЫХ СМЕСЕЙ, ИЗВЛЕЧЕНИЕ ПРИЗНАКОВ, КЕПСТРАЛЬНЫЕ КОЭФФИЦИЕНТЫ, ФОНЕМНАЯ СЕГМЕНТАЦИЯ

Объект исследования – цифровые записи речевых сигналов.

Предмет исследования – текстонезависимая автоматическая система идентификации человека по голосу.

Цель работы – разработка системы текстонезависимой идентификации диктора на основе моделей гауссовых смесей и тестирование ее работы на датасетах белорусского и английского языков.

Методы исследования – анализ существующих алгоритмов, изучение аналогов, спектральный анализ, аудиометод, моделирование, эксперимент, сравнение, изучение документации библиотек распознавания речи.

Результатом работы является приложение, реализующее компоненты системы текстонезависимой идентификации диктора на основе фонемной сегментации и моделей гауссовых смесей.

Области применения – контакт-центры, криминалистическая экспертиза, системы электронной торговли, охранные системы.

ABSTRACT

Master's thesis, 64 p., 38 fig., 3 tables, 28 sources, 3 appendices.

TEXT-INDEPENDENT SPEAKER IDENTIFICATION, BIOMETRY, GAUSSIAN MIXTURE MIXTURES, FEATURE EXTRACTION, CEPSTRAL COEFFICIENTS, PHONEME SEGMENTATION

Object of research - digital records of speech signals.

Subject of research - automatic text-independent system for speaker identification.

Purpose - development of text-independent speaker identification system based on Gaussian mixture models and testing it on belarusian and english datasets.

Research methods - analysis of existing algorithms, analysis of analogs, spectral analysis, audio method, modeling, experiment, comparison, analysis of documentation of speech recognition libraries.

The result of the work - an application that implements components of a text-independent speaker identification system based on phoneme segmentation and Gaussian mixture models.

Scope - contact centers, forensic expertise, electronic trading systems, security systems.

Уводзіны

У сувязі з інфарматызацыяй грамадства, пастаянным нарошчваннем патокаў перадачы даных па тэлефонных каналах сувязі, якія патрабуюць аўтэнтыфікацыі кліентаў, ростам значэння аўтаматычнай апрацоўкі дадзеных, а таксама нарастаючым выкарыстаннем аўтаматычных маўленчых тэхналогій задача ідэнтыфікацыі чалавека па голасе з'яўляецца адной з запатрабаваных сёння задач пазнавання вобразаў. Сістэмы распазнавання таго, хто гаворыць, па індывідуальных характарыстыках прамовы выклікаюць вялікую цікавасць як у навуковых, так і ў камерцыйных колах.

Нягледзячы на шматгадовыя даследаванні ў галіне маўленчых тэхналогій і зацікаўленасць як службаў бяспекі, так і бізнес-структур і шматлікіх катэгорый карыстальнікаў інфармацыйных паслуг, існуючыя сістэмы распазнавання дыктара далёкія ад дасканаласці і рэальнае іх прымяненне, за выключэннем вузкіх абласцей, досыць абмежавана. «Паводле рэгулярных гадавых справаздач Gartner Group, толькі каля аднаго працэнта аб'ёму патэнцыйных карыстальнікаў задаволена эфектыўнасцю камерцыйных сістэм распазнавання дыктара» [1]. Гэта тлумачыцца як тэарэтычнай складанасцю мадэлявання распазнавання чалавекам галасавога сігналу, так і наяўнасцю такіх праблем, як:

- залежнасць многіх характарыстык голасу ад эмацыйнага стану дыктара і стану яго здароўя;
- складаная акустычная абстаноўка (шумы і перашкоды);
- залежнасць вынікаў распазнавання ад характарыстык канала сувязі;
- ўзроставыя змены голасу;
- лёгкасць наўмыснага скажэння голасу да непазнавальнасці яго на слых чалавекам.

Павелічэнне колькасці кандыдатаў вымаўлення фразы, перашкоды ў канале перадачы сігналу, фонавыя шумы прымушаюць памыляцца нават чалавека, які лічыцца самым дасканалым на дадзены момант інструментам ідэнтыфікацыі іншага чалавека. Таму дасягненне стапрацэнтнай дакладнасці бачыцца задачай калі і выканальнай, то ў лабараторных умовах, якія не могуць улічыць велізарнай разнастайнасці фактараў, якія ўплываюць на якасць распазнавання голасу ў рэальным жыцці.

Разам з тым складанасць, шматузроўневасць працэсу фарміравання чалавечага маўлення дае неабмежаваныя магчымасці пошуку як яе характарыстык, якія робяць унікальнай гаворку кожнага чалавека, так і спосабаў камбінацыі гэтых характарыстык, якія ўлічваюць уплыў пералічаных вышэй фактараў, узаемадапаўняюць адзін аднаго і абцяжарваюць падробку голасу па адной-двух характэрных рысах. Выбар метадаў прыняцця рашэнняў пры гэтым таксама неабмежаваны.

Наяўнасць адкрытых баз запісаў маўлення мноства дыктараў значна спрыяе павелічэнню колькасці даследаванняў у сферы маўленчых тэхналогій і дае магчымасць аб'ектыўнага параўнання вынікаў, няхай і абмежаваную ў сувязі з рознымі патрабаваннямі сістэм распазнавання да фармату і колькасці ўваходных дадзеных. Але варта адзначыць, што такія базы прадстаўлены не для ўсіх моў у аднолькавай ступені, бо іх стварэнне – доўгая, карпатлівая праца.

Перыядычныя выпрабаванні на фіксаваных карпусах маўлення, якія арганізуюцца Нацыянальным інстытутам стандартаў і тэхналогій ЗША, дэманструюць паступовае павышэнне эфектыўнасці сістэм ідэнтыфікацыі дыктара [1], што дае стымул імкнуцца да дакладнасці канкурэнтных біяметрычных сістэм – па адбітках пальцаў, рысах твару, сятчатцы вока, малюнку вен або структуры генаў. Ідэнтыфікацыя па голасе выгадна адрозніваецца ад многіх з іх магчымасцю выкарыстання аддаленага доступу па тэлефоннай лініі. А пры актыўнай дзелавой дзейнасці карысна мець кругласутачны аператыўны доступ, напрыклад, да кіравання фінансавымі аперацыямі або банкаўскім рахункам.

Тое, што прадметам распазнавання ў сістэмах ідэнтыфікацыі чалавека па голасе з'яўляецца працэс, а не статычны здымак, дае ім дадатковую перавагу – магчымасць неабмежаванага павелічэння дакладнасці распазнання за кошт выкарыстання больш доўгіх фрагментаў прамовы. Гэтыя сістэмы незаменныя ва ўмовах, калі доступ не толькі аддалены, але і немагчыма атрымаць малюнак асобы карыстальніка – напрыклад, у цемры ці пры адсутнасці камеры. Разам з тым распазнаванне ўнікальных характарыстык голасу з'яўляецца добрым дадаткам да іншых сродкаў ідэнтыфікацыі ў выпадку магчымасці шматфактарнага аналізу.

Распазнаванне дыктараў, нават тэкстанезалежнае, для забеспячэння прымальнай дакладнасці звычайна патрабуе ведаў пра структуру мовы, на якой гаворыць дыктар, таму што вялікая частка інфармацыі пра яго асобу змяшчаецца ў тым, як ён прамаўляе словы і сказы – адзінкі мовы. Невыразныя гукі нясуць мала інфармацыі – толькі тэмбральную афарбоўку і мінімальныя межы дыяпазону частот голасу. Таму пры ўкараненні алгарытмаў распазнавання для канкрэтнай мовы пажадана дапрацоўка, якая ўлічвае яе асаблівасці.

Сёння вялікая частка сістэм распазнавання дыктараў арыентавана на ангельскую мову, існуюць распрацоўкі і для рускай – напрыклад, у расійскай кампаніі Маўленчыя Тэхналогіі. Але распазнаванне беларускіх дыктараў – задача пакуль мала даследаваная ў сувязі з меншай распаўсюджанасцю гэтай мовы. У дадзенай працы вырашана папоўніць гэты прабел і даследаваць прымяненне алгарытмаў распазнавання дыктараў у кантэксце беларускай мовы, а таксама напісаць сэрвіс, які рэалізуе адзін з гэтых алгарытмаў, і інтэрфейс для яго.

1. Пастаноўка задач

1.1. Пастаноўка асноўных задач

У ходзе выканання дадзенай працы неабходна даследаваць існуючыя алгарытмы распазнавання дыктараў і іх залежнасць ад мовы, рэалізаваць на іх аснове сістэму распазнавання дыктараў і прааналізаваць яе працаздольнасць на беларускім датасэце.

Для гэтага трэба вырашыць наступныя задачы:

1. Вывучыць існуючыя метады рашэння задачы распазнавання асобы па голасе, а таксама існуючыя сэрвісы апрацоўкі прамовы і магчымасці іх інтэграцыі.
2. Сабраць і падрыхтаваць дадзеныя для навучання сістэмы, правесці іх папярэднюю апрацоўку – фільтраванне, індэксацыю, прыдумаць спосабы аўтаматызацыі гэтай апрацоўкі і рэалізаваць іх.
3. Распрацаваць і рэалізаваць алгарытм ідэнтыфікацыі асобы па голасе:
    a. Апрацаваць гукавы сігнал і вылучыць істотныя для распазнання дыктара прыкметы;
    b. Пабудаваць мадэлі дыктараў на аснове атрыманых прыкмет;
    c. Выкарыстаць адзін з алгарытмаў прыняцця рашэнняў для вызначэння падабенства мадэляў з сігналам;
    d. Стварыць навучальныя і тэставыя выбаркі для тэставання атрыманага алгарытма;
    e. Даследаваць і выкарыстаць магчымасці паралельнай апрацоўкі атрыманых даных;
    f. Правесці эксперыментальнае даследаванне распрацаванага алгарытму і яго карэкціроўку ў адпаведнасці з атрыманымі вынікамі; у прыватнасці, аптымальным чынам наладзіць параметры алгарытма.

1.2. Мэты працы

Мэтамі дадзенай працы з'яўляюцца:

• вывучэнне працэсу фарміравання маўлення;

• аналіз метадаў здабывання характэрных прыкмет маўлення;

• аналіз спосабаў пабудовы мадэляў дыктара на аснове атрыманых прыкмет і іх залежнасці ад асаблівасцяў мовы;

• аналіз спосабаў прыняцця рашэнняў пры вызначэнні прыналежнасці запісанага адрэзка прамовы аднаму з дыктараў;

• аналіз існуючых сістэм ідэнтыфікацыі дыктараў, а таксама існуючых сэрвісаў апрацоўкі прамовы і магчымасці іх інтэграцыі;

• распрацоўка алгарытму тэкстанезалежнай ідэнтыфікацыі дыктараў;

• падрыхтоўка акустычных баз дадзеных для ангельскай і беларускай моў;

• рэалізацыя праграмы ў адпаведнасці з атрыманым алгарытмам і яе інтэграцыя ў адзін з існуючых сэрвісаў;

• ацэнка дакладнасці распазнання гэтай праграмай дыктараў на датасэтах розных моў (беларускай і ангельскай) і аналіз вынікаў.

2. Аналітычны агляд

2.1. Класіфікацыя сістэм распазнавання чалавека па голасе. Паняцце ідэнтыфікацыі дыктара

Задача ідэнтыфікацыі дыктара заключаецца ў вызначэнні на аснове унікальных характарыстык запісу прамовы чалавека, які яе вымавіў.

Па спосабе прыняцця рашэння вылучаюць наступныя падыходы да задач ідэнтыфікацыі чалавека па голасе:

- суб'ектыўныя, дзе рашэнне прымаецца экспертамі;
- аб'ектыўныя, мэтай якіх з'яўляецца выключэнне чалавечага фактару з працэсу прыняцця рашэння.

Да суб'ектыўным метадаў адносяцца распазнаванне дыктараў на слых і візуальнае параўнанне спектраграмы, аб'ектыўным спосабам ацэнкі з'яўляецца поўная аўтаматызацыя сістэм распазнання [1].

Па тыпу вырашаемай задачы сістэмы распазнавання чалавека па голасе дзеляцца на два віда:

- сістэмы ідэнтыфікацыі дыктара;
- сістэмы верыфікацыі дыктара.

Пры верыфікацыі атрыманы ўзор голасу дыктара параўноўваецца з эталонам і ўсталёўваецца яго ідэнтычнасць дадзенаму эталону. На аснове параўнання параметраў, унікальных для кожнага дыктара ў сістэме, прымаецца адно з двух рашэнняў: ці з'яўляецца дыктар тым, кім прадстаўляецца, ці не з'яўляецца.

Задача ідэнтыфікацыі дыктара «заключаецца ў выдзяленні з агульнай сукупнасці M дыктараў той асобы, якая па сваіх галасавых характарыстыках, заэталанаваным загадзя, будзе супадаць з характарыстыкамі голасу, які трэба апазнаць. Для ідэнтыфікацыі невядомай асобы запіс яго гаворкі павінен па чарзе параўноўвацца з эталоннымі запісамі ўсіх M магчымых дыктараў, у выніку чаго працэс ідэнтыфікацыі аказваецца залежным ад значэння M. Ад колькасці M аказваецца пры гэтым залежным велічыня верагоднасці памылкі ідэнтыфікацыі» [2, с. 110].

У залежнасці ад таго, ці хоча суб'ект быць апазнаным і прамаўляць парольную фразу, у сістэмах ідэнтыфікацыі дыктара выкарыстоўваюць ці не выкарыстоўваюць лексічную інфармацыю. Па выкарыстанні тэкставай інтэрпрэтацыі запісу сістэмы распазнання дыктара дзеляцца на:

- тэкстазалежныя сістэмы;

- тэкстанезалежныя сістэмы.

Якасць распазнавання ў текстазалежных сістэмах будзе заўсёды вышэй, чым у тэкстанезалежных, бо ў першым выпадку магчыма больш дакладная каліброўка ўваходных лексем з дапамогай ідэнтычнага маўленчага матэрыялу, а магчымасць кантролю гэтых лексем нярэдка распаўсюджваецца на прамоўцу і яго асяроддзе. У тэкстанезалежных сістэмах адсутнасць такіх магчымасцяў у нейкай меры кампенсуецца тым, што распазнаванне часта робіцца па больш доўгіх маўленчых фрагментах, а гэта дазваляе паляпшаць якасць ідэнтыфікацыі.

У сістэмах верыфікацыі магчымы індывідуальны падыход да падбору парольной фразы для кожнага дыктара з улікам яго артыкуляцыйных асаблівасцяў, а таксама стабілізацыя манеры вымаўлення зададзеных слоў карыстальнікам, у тым выпадку, калі кантрольная фраза падказваецца сістэмай вусна. Таксама вялікую ролю ў дакладнасці распазнавання выканае тое, што карыстальнік сам спрыяе правільнаму распазнаванню. Для сістэм ідэнтыфікацыі дыктара адну з галоўных цяжкасцяў уяўляе немагчымасць кантролю за ўсімі аспектамі задачы, а менавіта, выкарыстанне тэкстанезалежных зваротаў мовы, а таксама тое, што падазраваны ўсімі спосабамі імкнецца перашкодзіць правільнаму апазнанню. У гэтых сістэмах, акрамя выкарыстання сегментнай інфармацыі (асобных фанем і лексем), падключаюць да распазнавання і інфармацыю супрасегментную (рытм, тэмбр, мелодыка, часавыя характарыстыкі маўлення, сістэму націскаў), што ў вялікай ступені ўскладняе задачу ідэнтыфікацыі [3].

Акрамя таго, у залежнасці ад пастаноўкі задачы можна падзяліць сістэмы ідэнтыфікацыі на такія, што:

- працуюць на адкрытым мностве дыктараў;
- працуюць на закрытым мностве дыктараў.

Ідэнтыфікацыя на адкрытым мностве, у адрозненне ад закрытага, мае на ўвазе, што распазнаваны запіс можа не належаць ні аднаму з дыктараў, якія ўдзельнічаюць у навучанні сістэмы.

У дадзенай працы разгледжана тэкстанезалежная мадэль распазнавання дыктара ў сілу больш шырокага спектру яе прымянення, у прыватнасці, магчымасці незаўважнай і ненадакучлівай аўтаматычнай ідэнтыфікацыі кліентаў банкаў, аэрапортаў, клінік, кантактных цэнтраў і іншых арганізацый.

2.2. Асноўныя этапы вырашэння задачы распазнавання дыктараў

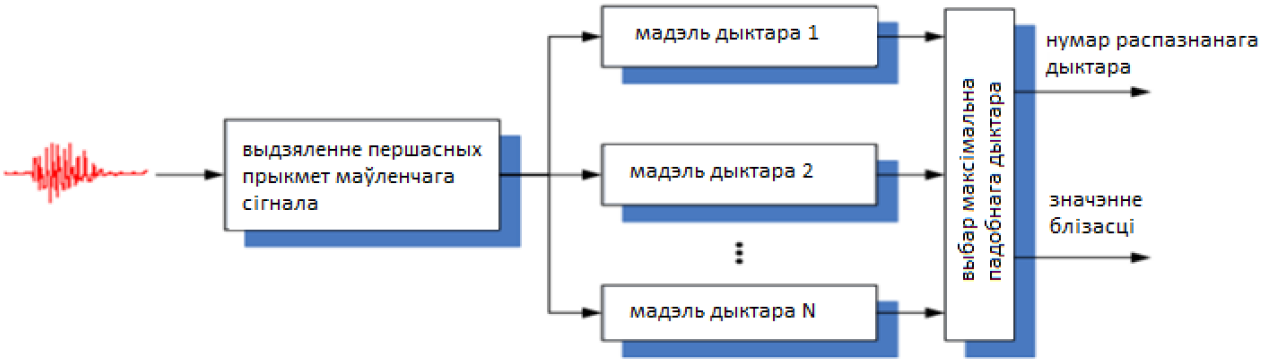
Працэс распазнавання дыктараў традыцыйна ўключае ў сябе наступныя асноўныя этапы:

1. Апрацоўка сігналаў. На гэтым узроўні вырашаецца задача 3.a: выбіраюцца важныя для распазнавання характарыстыкі маўлення; затым гэтыя характарыстыкі вылічаюцца з дапамогай розных пераўтварэнняў паслядоўнасці адлікаў ўваходных сігналаў.

2. Пабудова мадэляў. На гэтым узроўні вырашаецца задача 3.b: атрыманыя на першым этапе характарыстыкі выкарыстоўваюцца для пабудовы мадэляў дыктараў. «Мадэляванне можа заключацца як у простым капіраванні вектараў прыкмет, так і ў пабудове імавернасных мадэляў або іншых структур. Пасля гэтага становіцца магчымым вылічэнне ступені падабенства паміж прыкметамі і захаванай мадэллю» [4].

3. Прыняцце рашэння. На гэтым узроўні вырашаецца задача 3.c: атрыманыя на папярэднім узроўні мадэлі аналізуюцца на прадмет падобнасці на мадэль дыктара, які з'яўляецца аўтарам запісу. Гэты этап можа ўключаць параўнанне з некаторым парогам, вылічэнне ступеняў падабенства на ўзроўні мадэляў, верагоднасцяў адпаведнасці гэтых мадэляў мадэлі цікавага нам дыктара. Дыктар, якому адпавядае найбольш падобная па вызначаных крытэрыях мадэль, выбіраецца ў якасці выніку ідэнтыфікацыі.

На малюнку 2.1 прадстаўлена схема сістэмы ідэнтыфікацыі дыктара, якая ўключае вышэйпералічаныя этапы.



Малюнак 2.1 – Структурная схема сістэмы ідэнтыфікацыі

2.3. Атрыманне і перадапрацоўка запісаў

Аб'ектам распазнавання ў сістэмах ідэнтыфікацыі дыктара з'яўляюцца лічбавыя запісы чалавечай прамовы, якія прадстаўляюць яе сігнал паслядоўнасцю значэнняў амплітуд, атрыманых у выніку аблічбоўкі, напрыклад, з дапамогай імпульсна-кодавай мадуляцыі. «Для запісу і апрацоўкі маўленчага сігналу звычайна выкарыстоўваецца частата дыскрэтызацыі 8 або 16 кГц, больш высокая частата дыскрэтызацыі патрабуе вялікіх вылічальных выдаткаў» [4].

Прычынай зніжэння якасці распазнання звычайна з'яўляюцца фактары, звязаныя з запісам і перадачай сігналу, такія, як выкарыстанне розных каналаў

сувязі, розных мікрафонаў і адлегласцей да іх, дрэнная акустыка памяшкання. Напрыклад, пры распазнаванні голасу, які перадаецца па тэлефонным канале, неабходна ўлічваць уплыў перашкод і магчымае выкарыстанне розных мікрафонаў і каналаў перадачы сігналу пры ідэнтыфікацыі і рэгістрацыі прамоўцы. Папярэдняя апрацоўка сігналу можа заключацца ў выдаленні участкаў, дзе няма маўлення, выдаленні пастаяннага амплітуднага зрушэння, дызерінге, апрацоўцы частотным фільтрам.

Выманню нізкаўзроўневых спектральных прыкмет абавязкова папярэднічае разбіццё ўваходнай лічбавай паслядоўнасці сігналу на адрэзкі (вокны) даўжынёй каля 20-25 мс. Вядома, што пры выкарыстанні дыскрэтнага пераўтварэння Фур'е для знаходжання спектру сігналу ўзнікаюць цяжкасці, абумоўленыя канечнасцю інтэрвалу апрацоўкі. Таму кожны адрэзак дамнажаецца на адну з аконных функцый, якія візуальна паляпшаюць частотны спектр на разрыве межаў акна шляхам падаўлення яго бакавых пялёсткаў.

Нярэдка пры апрацоўцы сігналу ўжываюць павелічэнне амплітуды яго высокачастотных кампанент (англ. Pre-emphasis) з прычыны іх большай інфарматыўнасці пры працы з чалавечым маўленнем.

2.4. Метады здабывання прыкмет маўлення

Разгледзім задачу 3.а – вылучэнне параметраў, якія адрозніваюць гаворку аднаго чалавека ад гаворкі другога. За дзясяткі гадоў было знойдзена мноства такіх прыкмет, ад фізіялагічных, абумоўленых анатамічнымі асаблівасцямі маўленчага тракту чалавека, і да паводзінных, якія змяняюцца з часам, узростам, сацыяльным асяроддзем. Традыцыйна сістэмы тэкстанезалежнай ідэнтыфікацыі дыктара робяць упор на выкарыстанні фізіялагічных прыкмет у сілу іх большай сталасці і незалежнасці ад сэнсу сказанага, але спалучэнне іх з набытымі характарыстыкамі можа станоўча адбіцца на дакладнасці ідэнтыфікацыі ў сувязі з павелічэннем ўстойлівасці да перашкод канала сувязі, што неаднаразова пацвярджалася эксперыментальна [5].

2.4.1. Фізіялагічныя характарыстыкі маўлення. Фарміраванне маўленчага сігналу

Фарміраванне акустычнага маўленчага вагання, які перадае думку дыктара, заключаецца ў пераўтварэнні паслядоўнасці нервовых імпульсаў з дапамогай артыкуляцыйнага апарата. На кожным этапе гэтага працэсу фармуюцца індывідуальныя прыкметы дыктара:

1. З лёгкіх выштурхваецца бруя паветра, якая з'яўляецца асновай фарміравання гучання. «У залежнасці ад ёмістасці лёгкіх і характару дыхання (ключычны, грудны ці брушны тып) чалавек пры гэтым робіць неаднолькавую колькасць удыхаў і выдыхаў і затрачвае на іх розны час» [2, с. 15]. Гэта вызначае цыкл маўленчага дыхання, рытмічную структуру маўлення і сілу гучання.

2. Паветраны струмень пачынае вібраваць, праходзячы праз галасавыя звязкі, размешчаныя ў гартані. Ад даўжыні, таўшчыні і ступені нацяжэння галасавых звязкаў залежыць характар вагання галасавых звязкаў, вышыня голасу, яго сіла і тэмбр, але апошнія дзве характарыстыкі значна відазмяняюцца ў далейшым. [2, с. 17-18].

3. Вібрацыя ў бруі паветра знаходзіць асаблівую форму дзякуючы рэзанатару, сфармаванаму ў глотцы, ротавай і насавой паражнінах органамі артыкуляцыі. Тут голасу надаецца індывідуальная тэмбравая афарбоўка дзякуючы рэгуляцыі чалавекам узаемаразмяшчэнняў рухомых (мова, вусны, увула) і нерухомых (мяккае і цвёрдае неба, зубы) органаў. [6].

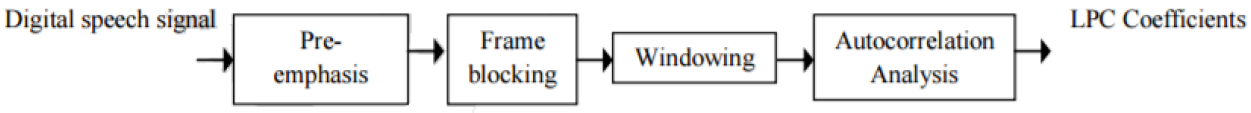
Такім чынам, індывідуальнасць голасу вызначаецца анатоміяй лёгкіх, галасавых складак, маўленчага тракта і артыкуляцыйнага апарата. «Частата ваганняў зморшчын і форма імпульсаў аб'ёмнай хуткасці патоку, які праходзіць праз галасавую шчыліну, уплываюць на форму абгінаючай спектру маўленчага сігналу і яго часовыя параметры. Геаметрычныя памеры розных аддзелаў маўленчага тракта і бакавыя паражніны (грушападобныя паражніны ў галіне гартані, дзве насавыя паражніны, гайморовы паражніны), а таксама механічныя ўласцівасці тканін маўленчага тракта вызначаюць яго рэзанансныя частоты і хуткасць згасання ваганняў на рэзанансных частотах. У спектры маўленчага сігналу гэта праяўляецца як шырыня і частоты яго пікаў» [1].

Калі казаць пра індывідуальнасць прамовы, то да фізічных асаблівасцяў голасу дадаюцца паводніцкія, псіхалагічна абумоўленыя асаблівасці гаворкі, такія, як выкарыстанне пэўных канструкцый і зваротаў мовы.

2.4.2. Каэфіцыенты на аснове лінейнага прадказання

Форма маўленчай хвалі адназначна вызначаецца крыніцай – галасавой шчылінай у гартані – і фільтрам – органамі маўленчага тракта. Дадзены факт выкарыстоўваецца пры кадаванні гаворкі з дапамогай лінейнага прадказання (англ. LPC, Linear Predictive Coding): маўленне разглядаецца як сігнал на выхадзе дыскрэтнага фільтра, якім апраксімуецца маўленчы тракт, і па гэтым сігнале шукаецца зыходны сігнал і параметры гэтага фільтра.

Лінейнае прадказанне дазваляе апраксімаваць бягучы адлік ўзважанай камбінацыяй папярэдніх.



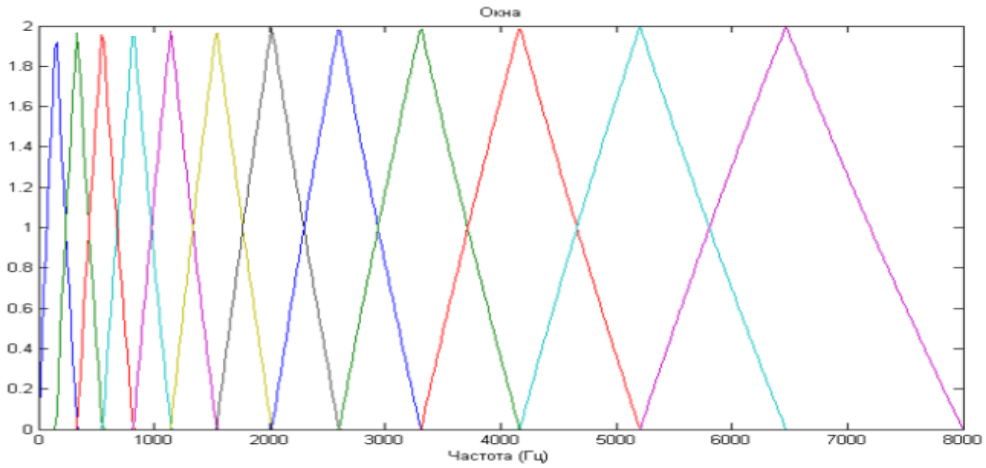
Малюнак 2.2 – Блок-схема атрымання LPC-каэфіцыентаў

У сучасных сістэмах распазнавання маўлення параметры лінейнага прадказання не выкарыстоўваюцца наўпрост, бо маюць велізарны дынамічны дыяпазон, вельмі адчувальныя да частот уваходнага сігналу і моцна карэляваныя паміж сабой [7], але выкарыстоўваюцца пры вылічэнні вытворных мадыфікаваных прыкмет.

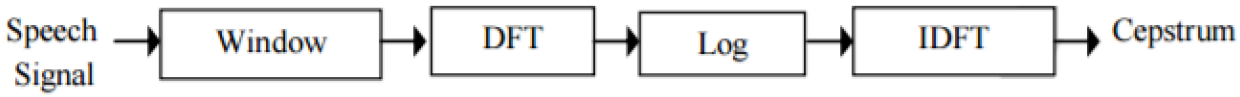
2.4.3. Кепстральны аналіз

Кепстральны аналіз мадэлюе размеркаванне спектральнай энергіі сігналу шляхам знаходжання кепстра – спектру ад спектру, які атрымліваецца зваротным пераўтварэннем Фур'е лагарыфма спектру магутнасці зыходнага сігналу. Гэтая магутнасць можа вылічвацца на аснове LPC-каэфіцыентаў – тады атрымліваюцца кепстральные каэфіцыенты лінейнага прадказання (англ. LPCC, Linear Prediction Cepstral Coefficients) – або з дапамогай хуткага пераўтварэння Фур'е. Звычайна яно ўжываецца на лагарыфмічнай шкале, напрыклад, Мел (англ. Mel Scale) або Барк (англ. Bark Scale), бо ўспрыманне чалавекам вышыні гуку залежыць ад яго частоты не лінейна, а лагарыфмічна.

Так, напрыклад, фільтры мел-шкалы размешчаны лінейна на нізкіх частотах і лагарыфмічна – на высокіх:



Малюнак 2.3 – Грабянец фільтраў мел-шкалы



Малюнак 2.4 – Блок-схема атрымання кепстральных каэфіцыентаў

Выкарыстанне шкал частот і паласавых фільтраў абумоўлена існаваннем крытычных палос, такіх, што частоты ў межах паласы неадметныя на слых для чалавечага вуха.

Каэфіцыенты кепстра знаходзяцца па формулах

$$c_n = \frac{1}{\Theta}\int_0^{\Theta} \log|S(jw,t)|^2\, e^{-jn\Omega w} dw \quad (2.1)$$

$\Omega = \frac{2\pi}{\Theta}, \Theta$ – верхняя частата ў спектры маўленчага сігналу, $|S(jw,t)|^2$ – спектр магутнасці. Калі выкарыстоўваецца грабянец паласавых фільтраў, то выкарыстоўваецца дыскрэтнае кепстральное пераўтварэнне, каэфіцыенты якога вылічаюцца як

$$c_n = \sum_{m=1}^{M}[\log Y(m)] \cos\left[\frac{\pi n}{M}(m-\frac{1}{2})\right] \quad (2.2)$$

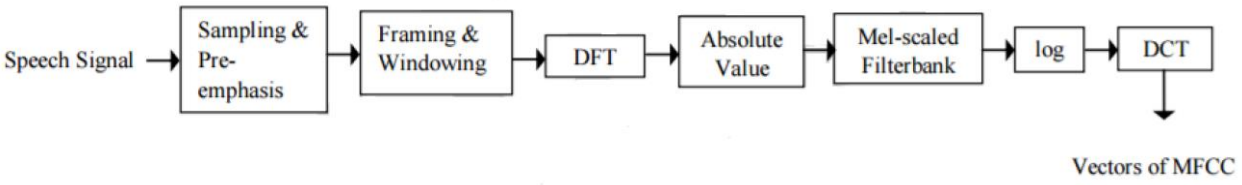
дзе Y(m) – выхадны сігнал m-га фільтра, c_n – n-ы каэфіцыент кепстра.

Кепстр апісвае форму абгінаючай спектру сігналу, у якой інтэгруюцца характарыстыкі крыніц ўзбуджэння і формы маўленчага тракта.

У эксперыментах па распазнаванню голасу людзьмі на слых быў ўстаноўлены моцны ўплыў абгінаючай спектру на пазнавальнасць голасу [1]. Таму выкарыстанне таго ці іншага спосабу аналізу абгінаючай спектру ў мэтах распазнання дыктара апраўдана.

2.4.3.1. Мел-частотныя кепстральныя каэфіцыенты

Мел-шкала выкарыстоўваецца пры вылічэнні мел-частотных кепстральных каэфіцыентаў (англ. MFCC, Mel-Frequency Cepstral Coefficients) [8], якія сталі стандартам у галіне маўленчых тэхналогій дзякуючы адноснай прастаце вылічэння і добрай здольнасці да апраксімацыі. Схема іх вылічэння прадстаўлена на малюнку 2.5.



Малюнак 2.5 – Блок-схема атрымання мел-частотных кепстральных каэфіцыентаў

Першыя каэфіцыенты кепстра нясуць інфармацыю пра ўзровень ўзбуджэння крыніцы, а апошнія – пра вакальны тракт. Таму пры выкарыстанні MFCC часам можа не ўлічвацца першы каэфіцыент, які з'яўляецца мерай агульнай інтэнсіўнасці сігналу.

Існуюць мадыфікацыі MFCC, напрыклад, HFCC (англ. Human Factor Cepstral Coefficients), дзе замест мел-шкалы выкарыстоўваецца шкала Human-factor, або TMFCC (англ. Teager MFCC), але яны пакуль мала даследаваны.

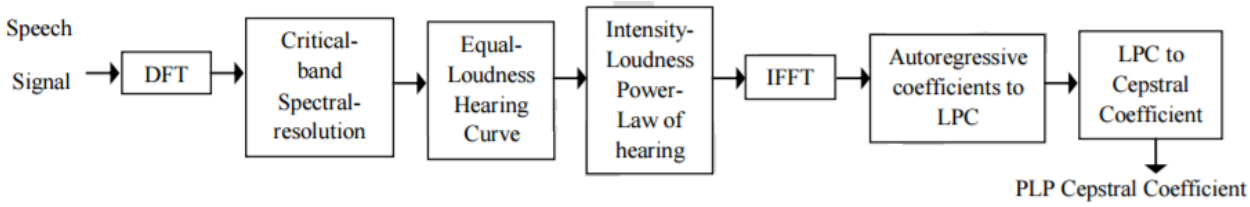
Недахопам мел-частотных каэфіцыентаў у кантэксце распазнавання дыктара з'яўляецца тое, што гэты метад забяспечвае добрую адрознівальную здольнасць у нізкачастотнай вобласці і нізкую ў высокачастотнай. Найбольшая інфармацыя пра дыктара знаходзіцца ў палосах ніжэй 600 Гц і вышэй 3000 Гц,

таму частка карыснай інфармацыі ў галіне высокіх частот можа дрэнна ўлічвацца.

Але гэты недахоп можна кампенсаваць знаходжаннем зваротных каэфіцыентаў – IMFCC (англ. Inverse MFCC) [9], дзе фільтры мел-шкалы размешчаны люстэркава: лінейна на высокіх частотах і лагарыфмічна – на нізкіх. Яны, наадварот, успрымальныя да інфармацыі высокіх частот.

2.4.3.2. Каэфіцыенты перцэпцыйнага лінейнага прадказання

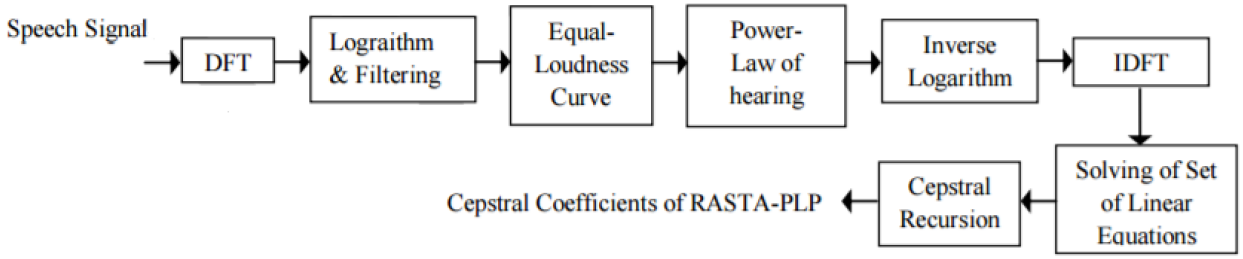
Каэфіцыенты перцэпцыйнага лінейнага прадказання (англ. PLP, Perceptual Linear Prediction) [10] заснаваны на вылічэнні кароткачасовых спектраў, пераўтварэнні іх некалькімі псіхафізічна абумоўленымі спектральнымі пераўтварэннямі, вылічэнні значэнняў іх аўтакарэляцыі і выкарыстанні іх у якасці каэфіцыентаў лінейнага прадказання з наступным вылічэннем кепстральных каэфіцыентаў:



Малюнак 2.6 – Блок-схема атрымання перцэпцыйных лінейных кепстральных каэфіцыентаў

Каэфіцыенты PLP зручна выкарыстоўваць у сістэмах аўтаматычнага распазнавання маўлення, бо яны паказваюць дастаткова моцную залежнасць ад зместу тэксту.

Па той прычыне, што PLP-каэфіцыенты моцна схільныя да перашкод у канале перадачы дадзеных і паказваюць значнае памяншэнне эфектыўнасці пры аналізе маўлення ў рэальным зашумленым асяроддзі, іх часам камбінуюць з RASTA-фільтраваннем (англ. Relative Spectral Transform), якое абапіраецца на меркаванне пра стацыянарнасць характарыстык канала і выкарыстоўвае палосна-прапускаючы рэгрэсійны фільтр лагарыфма спектру для падаўлення шумоў і аднімання пастаяннай кампаненты спектру (гл. малюнак 2.7).



Малюнак 2.7 – Блок-схема атрымання RASTA-PLP каэфіцыентаў

2.4.4. Каэфіцыенты вышэйшых парадкаў. Камбінаванне спектральных каэфіцыентаў

Апісаныя вышэй метады прызначаны для здабывання прыкмет на невялікіх неўпарадкаваных адрэзках маўлення і не выкарыстоўваюць інфармацыю пра яе дынаміку. Таму часам вектары прыкмет аб'ядноўваюць з іх першымі або нават другімі вытворнымі – рознасцямі па часе, якія атрымалі назву дэльта- і дэльта-дэльта-каэфіцыентаў.

Вытворныя параметры, з аднаго боку, могуць палепшыць вынікі распазнавання, з другога – моцна карэляваць з каэфіцыентамі першага парадку і ў N раз павялічваюць памер атрыманага вектара прыкмет, дзе N – парадак вытворных.

Таксама дастаткова моцна карэлююць паміж сабой каэфіцыенты MFCC і LPCC, бо абедзве гэтыя прыкметы апісваюць абгінаючую спектру; крыху менш – PLP. Таму камбінацыя іх хоць і можа прыводзіць да павышэння дакладнасці распазнання, але мае вялікую долю лішкавасці і можа быць прыменена толькі калі залішні рэсурсы памяці і прадукцыйнасці сістэмы.

2.4.5. Высокаўзроўневыя прыкметы

Чым даўжэй доследныя адрэзкі маўлення, тым на больш высокім узроўні можна разглядаць маўленне, ад разгледжаных раней спектральных каэфіцыентаў да аналізу сэнсу сказанага. Градацыя прыкмет маўлення па ўзроўням прадстаўлена на малюнку 2.8:

Малюнак 2.8 – Тыпы характарыстык маўлення

Перавагай прыкмет самага нізкага ўзроўню, што вылічаюцца на кароткіх адрэзках прамовы, у кантэксце ідэнтыфікацыі дыктара з'яўляецца іх абумоўленасць пабудовай маўленчага тракта, і, як следства, незалежнасць ад сэнсавай нагрузкі. Але ў сілу кароткачасовасці аналізу такія прыкметы бываюць адчувальныя да перашкод у канале перадачы сігналу і фонавага шуму.

Прыкметы больш высокіх узроўняў адлюстроўваюць паводніцкія, сацыяльна і псіхалагічна абумоўленыя набытыя характарыстыкі маўлення. Яны больш устойлівыя да скажэнняў канала, чым фізіялагічныя прыкметы, але маюць свае недахопы, у тым ліку складанасць здабывання і патрабавальнасць да аб'ёму і разнастайнасці неабходных для гэтага дадзеных.

Да прасадычных характарыстык маўлення адносяць дынаміку частаты асноўнага тону, інтэнсіўнасць, націск, інтанацыю, працягласць і частату паўз і фанетычных сегментаў, індывідуальныя эфекты каартыкуляцыі. У адрозненне ад сегментных адзінак прамовы (гукі, фанемы, словы), гэтыя характарыстыкі напластоўваюцца на ланцужок сегментаў, закранаючы і склады, і словы, і сказы. Многія з іх дрэнна падыходзяць для задачы ідэнтыфікацыі дыктара ў сілу сваёй моцнай залежнасці ад кантэксту – настрою і намераў дыктара, сітуацыі, зместу прамовы.

На ўзроўні фанем разглядаюць, напрыклад, фармантныя частоты галосных. Фарманта з'яўляецца выразнай вобласцю узмоцненых частот у дыяпазоне гуку, у якой па прычыне рэзанансу ўзмацняецца некаторы лік гарамонік тона, вырабленага галасавымі звязкамі. Колькасць фармант супастаўна з колькасцю рэзанансных паражнін ў маўленчым тракце, а іх частоты залежаць ад канфігурацыі і памераў галасавога тракта, што абумоўлівае іх ужыванне ў сістэмах распазнання дыктара. Перавагай фармант з'яўляецца тое, што яны захоўваюцца нават пры вымаўленні паведамлення шэптам. Фармантныя характарыстыкі голасу практычна немагчыма падрабіць з мэтай імітацыі, але нескладана змяніць з мэтай маскіроўкі, таму пажадана выкарыстоўваць іх у спалучэнні з іншымі прыкметамі.

Улік асаблівасцяў вымаўлення асобных галосных і зычных фанем, асабліва паталагічных, відавочна, мае месца ў распазнаванні чалавечай прамовы, асабліва чалавекам. Пры наўмыснай імітацыі вымаўлення дыктара імітатар у першую чаргу спрабуе пераняць шапялявасць, картавасць і іншыя асаблівасці вымовы асобных фанем. У сістэмах аналізу маўлення гэтыя асаблівасці вылічваюцца на аснове розных ацэнак размеркавання энергіі традыцыйных спектральных малюнкаў, зробленых праз пэўныя інтэрвалы на ўсёй працягласці гуку.

Лінгвістычныя прыкметы часцей выкарыстоўваюцца ў сістэмах распазнання маўлення, чым у сістэмах распазнання дыктара, бо для фарміравання статыстык распаўсюджанасці і ўзаемнага размяшчэння слоў патрэбна выбарка запісаў з максімальна магчымым лексіконам, а пры навучанні запісаў аднаго дыктара аб'ём выкарыстанага слоўніка бывае недастаткова вялікі. Такая інфармацыя больш прыдатная для вызначэння мовы і дыялекту дыктара, чым для яго непасрэднай ідэнтыфікацыі, хоць некаторых дыктараў і можна вызначыць па словах-паразітах. Семантычны аналіз таксама больш прыдатны

для задач распазнавання маўлення, чым задач распазнавання дыктара, бо залежыць ад намераў прамоўцы не менш, чым абраныя ім словы.

2.5. Мадэлі дыктараў і вырашальныя правілы

На этапе пабудовы мадэляў дыктара, гэта значыць рашэння задачы 3.b, неабходна набор прыкмет для дыктара прадставіць у выглядзе, прыдатным для далейшага параўнання і аналізу і ў канчатковым выніку прыняцця рашэння (рашэння задачы 3.c). Пажадана, каб для мадэлі быў вызначаны спосаб знаходжання ступені падабенства яе іншым мадэлям і распазнаваным узорам, напрыклад, на аснове адной з метрык (такой, як эўклідавая адлегласць паміж вектарамі) або на аснове ацэнкі верагоднасці. Па той прычыне, што пры распазнаванні па голасе, у адрозненне ад многіх біяметрычных сістэм, прадметам распазнавання з'яўляецца працэс, а не статычны малюнак, мадэль дыктара часта ўяўляецца ў выглядзе паслядоўнасці вектараў [4]. Кожны вектар адпавядае вызначанаму ўчастку маўленчага сігналу, а парадак вектараў захоўвае парадак гэтых участкаў ў часе. Тым не менш, выкарыстанне часовай інфармацыі неабавязкова і залежыць ад выбару мадэлі.

Існуе некалькі спосабаў класіфікацыі мадэляў для задачы распазнавання. У літаратуры часта вылучаюць мадэлі:

- генератыўныя;
- дыскрымінатыўныя.

Сутнасць генератыўных мадэляў заключаецца ў мадэляванні дадзеных, атрыманых для навучання, напрыклад, з дапамогай ацэнкі функцыі шчыльнасці верагоднасці. Прыкладам можа служыць мадэль гаусавых сумесяў. Дыскрымінатыўныя мадэлі заснаваны на пабудове мяжы паміж класамі, як гэта рэалізавана, напрыклад, у метадзе апорных вектараў.

2.5.1. Метады бліжэйшых суседзяў

Няхай маецца N дыктараў, аналізуемы запіс прадстаўлены як паслядоўнасць з L вектараў, мноства вектараў паслядоўнасці дыктара C_j абазначым S_j, $S_j \in S$. Пры выкарыстанні метаду бліжэйшых суседзяў вылічаецца ступень падабенства кожнага вектара тэставай паслядоўнасці y_i, $i \in \overline{1,L}$, і кожнага вектара $x_p \in S$, кожнага шаблона дыктара C_j. Сярод усіх шаблонных вектараў выбіраюцца k максімальна падобных вектараў $x_1, x_2, \dots, x_k$. Пры k=1 дадзены метад называюць метадам бліжэйшага суседа. У гэтым выпадку верагоднасць прыналежнасці вэктару y_i класу (дыктару) C_j

$$\hat{P}(C_j|y_i) = \begin{cases} 1, & j = \arg\max\limits_{x_p \in S_j} f(y_i, x_p) \\ 0, & \text{інакш} \end{cases} \quad (2.3)$$

дзе $f(x, y)$ – ступень падабенства вектараў.

Няхай k > 1 і k_{ij} – колькасць вектараў, якія належаць класу C_j, сярод знойдзеных k суседзяў. Тады вектар y_i належыць класу C_j з верагоднасцю

$$\hat{P}(C_j|y_i) = \frac{k_{ij}}{k} \tag{2.4}$$

дзе $\sum_{j=1}^{N} P(C_j|y_i) = 1$. У агульным выпадку паслядоўнасць вектараў класіфікуецца па схеме галасавання:

$$\mathrm{C} = \arg\max_{1\le j\le N} \sum_{i=1}^{L} \hat{P}(C_j|y_i) = \arg\max_{1\le j\le N} \sum_{i=1}^{L} k_{ij} \tag{2.5}$$

Ва ўзважаным варыянце гэтага алгарытму пры разліку $\hat{P}(C_j|y_i)$ ступені падабенства кожнага з суседзяў улічваюцца як вагі. Гэта значыць, што ўлічваецца не толькі колькасць вектараў, якія належаць пэўнаму дыктару, але і ступень падабенства кожнага з іх тэставаму вектару.

Недахопам гэтых метадаў з'яўляецца неабходнасць захавання ўсёй паслядоўнасці навучальных вектараў. Для зніжэння працаёмкасці вылічэнняў выкарыстоўваюць розныя метады скарачэння памернасці мадэлі дыктара або метады захавання дадзеных для паскарэння пошуку, такія, як, напрыклад, k-мернае дрэва.

2.5.2. Вектарнае квантаванне

У метадзе вектарнага квантавання, у адрозненне ад метаду бліжэйшага суседа, мноства навучальных вектараў захоўваецца не цалкам, а пераўтвараецца ў мноства кодавых вектараў звычайна фіксаванага памеру. Алгарытм k-сярэдніх [11] з'яўляецца распаўсюджаным метадам пабудовы такога мноства, названага таксама кодавай кнігай. Атрыманыя кодавыя вектары выкарыстоўваюцца для пабудовы шаблону, а вылічэнне адлегласці паміж ўваходнай паслядоўнасцю вектараў і кодавымі кнігамі ажыццяўляецца аналагічна метаду бліжэйшых суседзяў.

Вектарнае квантаванне ў такім выглядзе, як і метад бліжэйшых суседзяў, не абапіраецца на фанетычныя элементы, з-за чаго ў сістэмах, незалежных ад кантэксту, будзе ўзнікаць рознагалоссе паміж навучальным і распазнаваным кантэкстам. Таму гэтыя метады можна выкарыстоўваць толькі ў простых тэкстазалежных сістэмах распазнання дыктара па ключавым слове.

2.5.2. Мадэль гаусавых сумесяў

Мадэль гаусавых сумесяў (англ. GMM, Gaussian Mixture Model) шырока выкарыстоўваецца ў галіне распазнання дыктараў. Яна ўяўляе сабой узважаную суму гаусіан

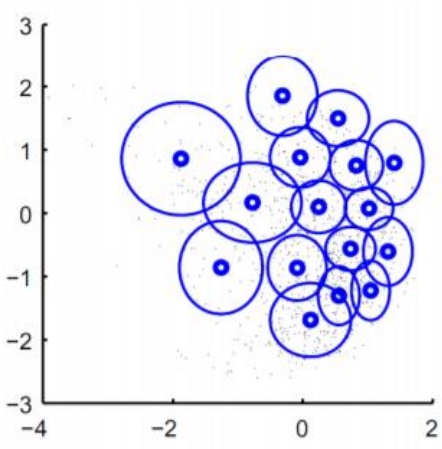
$$p(x|\lambda) \sum_{i=1}^{M} w_i\, p_i(x) \tag{2.6}$$

дзе λ – мадэль дыктара, M – колькасць яе кампанентаў, w_i – іх вагі, якія ў суме складаюць адзінку. Функцыя шчыльнасці верагоднасці кожнага кампанента задаецца формулай

$$p_i(x) = \frac{1}{(2\pi)^{D/2}|\Sigma_i|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu_i)^T \sum_i^{-1}(x-\mu_i)\right) \quad (2.7)$$

дзе D – памернасць прасторы прыкмет, μ_i – вектар матэматычнага спадзявання, Σ – матрыца каварыяцыі. Часцей за ўсё ў сістэмах, якія рэалізуюць дадзеную мадэль, выкарыстоўваецца дыяганальная матрыца каварыяцыі. Магчыма таксама выкарыстанне адной матрыцы каварыяцыі для ўсіх кампанентаў мадэлі дыктара або адной матрыцы для ўсіх мадэляў. Такім чынам, для пабудовы мадэлі дыктара неабходна вызначыць вектары сярэдніх, матрыцы каварыяцыі і вагі кампанентаў. Дадзеную задачу вырашаюць з дапамогай алгарытму максімізацыі спадзявання (англ. Expectation Maximization) [12] ці, у некаторых выпадках, з дапамогай больш вылічальна эфектыўнага алгарытму Вітэрбі [13].

У адрозненне ад вектарнага квантавання, мадэль гаусавых сумесяў выкарыстоўвае вобласці ў прасторы прыкмет, якія перакрываюцца паміж сабой:



Малюнак 2.9 – Апраксімацыя размеркавання сумессю нармальных размеркаванняў

Мадэлі гаусавых сумесяў можна будаваць на розных узроўнях. На самым нізкім мадэль будуецца па ўсіх фрэймах усіх запісаў дыктара, прадстаўляючы сабой нейкі ўсярэднены варыянт вымаўлення ім любога гуку – набліжэнне даволі грубае. На больш высокіх маўленне дыктара разбіваецца на класы (гэта могуць быць, напрыклад, гукі) і для кожнага з гэтых класаў будуецца свая мадэль. На самым высокім будуюцца мадэлі мадэляў: характарыстыкі маўлення дыктара, для якога будуецца мадэль, прадстаўляюцца як сумесь характарыстык маўлення дыктараў складзенай загадзя рэферэнтнай базы. Каэфіцыенты атрыманай сумесі з'яўляюцца ўнікальнымі для кожнага дыктара і выкарыстоўваюцца ў якасці яго мадэлі.

У [14] GMM паказалі найлепшыя вынікі пры вырашэнні задачы ідэнтыфікацыі дыктара ў параўнанні з метадам апорных вектараў і алгарытмам дынамічнай трансфармацыі часовай шкалы. У прыватнасці, таму яны выкарыстаны пры пабудове мадэляў у распрацаванай сістэме ідэнтыфікацыі.

2.5.3. Метад апорных вектараў

Метад апорных вектараў (англ. SVM, Support Vector Machines) дазваляе пабудаваць гіперплоскасць у шматмернай прасторы, якая падзяляе два класы, напрыклад, параметраў мэтавага дыктара і параметраў дыктараў з рэферэнтнай базы. Поспех яго прымянення залежыць ад таго, наколькі ўдала падабрана нелінейнае пераўтварэнне прасторы вымераных параметраў у кожным канкрэтным выпадку пры распазнанні дыктараў.

Метад апорных вектараў можа прымяняцца ў камбінацыі з метадам GMM або метадам схаваных маркаўскіх мадэляў (англ. HMM, Hidden Markov Models) [15], зарэкамендаваў сябе ў задачах аўтаматычнага распазнавання маўлення.

2.5.4. Нейронавыя сеткі

Канкрэтная мадэль у гэтым выпадку будуецца асобна для кожнай задачы на аснове існуючых мадэляў (такіх, як шматслойныя персептроны, рэкурэнтныя сеткі, сеткі на аснове радыяльна-базісных функцый, сеткі прамога распаўсюджвання, свёртачныя сеткі і іншыя). Недахопам гэтага падыходу з'яўляецца тое, што поўны працэс навучання можа патрабаваць вялікіх часавых і вылічальных выдаткаў. Патрабуецца падбіраць шмат характарыстык сеткі і параметраў навучання, і гэты выбар невідавочны. Тым не менш нейронавыя сеткі выкарыстаны ў многіх работах, яны гнуткія і могуць быць арыентаваны на рашэнне канкрэтнай праблемы распазнавання, напрыклад, праблемы наяўнасці значных акустычных шумоў. Перспектыўным напрамкам сёння з'яўляецца выкарыстанне глыбокіх нейронавых сетак (англ. Deep Neural Networks) у сферы распазнання маўлення [16], як і пры вырашэнні задач распазнавання вобразаў у цэлым.

2.5.5. I-vectors

У нядаўніх даследаваннях і публікацыях [25], [26] папулярнасць набірае мадэляванне характарыстык дыктара з дапамогай так званых i-vectors. Іх знаходжанне ўяўляе сабой працэс імавернаснага сціску, які паменшвае памернасць супервектараў у адпаведнасці з лінейнай гаусавай мадэллю. Супервектар s медыян мадэляў гаусавых сумесяў зыходнага запісу праецыруецца на прастору нізкай памернасці, названую Total Variability Space:

$$s = m + Tw \tag{2.8}$$

дзе T – матрыца Total Variability (англ. Total Variability matrix), m – незалежны ад дыктара супервектар медыян універсальнай фонавай мадэлі (англ. UBM, Universal Background Model), w – залежны ад канала і дыктара i-vector.

Іх з'яўленне стала развіццём сумеснага фактарнага аналізу (англ. JFA, Joint Factor Analysis) GMM-мадэляў, у якіх аналізуемы вектар запісу раскладваюць на кампаненты, залежныя ад дыктара, незалежныя ад дыктара, залежныя ад канала і рэшткавыя, пасля таго, як было заўважана, што залежныя ад канала кампаненты таксама нясуць інфармацыю пра дыктара. У JFA выкарыстоўваецца, магчыма, занадта моцная здагадка пра фактарызацыю і незалежнасць гэтых кампанент. Пры ідэнтыфікацыі дыктара з дапамогай i-vectors інфармацыю пра канал кампенсуюць часцей за ўсё з дапамогай PLDA (англ. Probability Linear Discriminant Analysis), якая дазваляе карэляцыю паміж кампанентамі.

Адным з недахопаў i-vectors лічыцца іх нестабільнасьць у розных умовах запісу, адчувальнасць да змены канала.

2.6. Аналіз рынку сістэм ідэнтыфікацыі дыктара

У апошнія гады назіраецца павелічэнне попыту на паслугі галасавой біяметрыі ў такіх сферах, як мабільны банкінг, судова-медыцынская экспертыза, ахова здароўя, сістэмы дзяржаўнай бяспекі.

Лідэрам на сусветным рынку ў сферы распазнання маўлення, якая займае больш за палову яго долі, з'яўляецца транснацыянальная карпарацыя Nuance Communications (распрацоўшчык інтэрактыўнага сэрвісу Siri кампаніі Apple). Яна распрацоўвае праграмы ў многіх сумежных галінах: для мабільнай верыфікацыі, праваахоўных органаў, экспертнага аналізу галасавых запісаў, ідэнтыфікацыі банкаўскіх кліентаў (Nuance Forensics, Nuance Identifier, Nuance Mobile Identifier і інш.). На расійскім рынку лідэрам можна лічыць кампанію Маўленчыя тэхналогіі, якая зрабіла платформу галасавой ідэнтыфікацыі і верыфікацыі VoiceKey, укаранёную, напрыклад, у ААТ «Прыорбанк», і сістэму крыміналістычнага ўліку і біяметрычнага пошуку па голасе і твары VoiceGrid. Сярод іншых рашэнняў можна згадаць сістэму KIVOX Passive Detection і Voice ID амерыканскай кампаніі Agnitio, а таксама Voiceprint Recognition Engine кітайскай кампаніі IFLYTEK. Але прадукты гэтых кампаній пакуль досыць дарагія, і не ўсе прадпрыемствы і ўстановы дазваляюць сабе іх выкарыстоўваць.

Менш вядомыя прадукты часта апісаны павярхоўна, што не дазваляе быць упэўненымі ў іх якасці (3I SpeakerID кампаніі 3I Technologies, сістэма мульцібіяметрычнай ідэнтыфікацыі BioLink AMIS кампаніі BioLink), альбо маюць шэраг недахопаў. Тэхналогія GritTec's Speaker-ID расійскай кампаніі GritTec забяспечвае дакладнасць ідэнтыфікацыі толькі 85% пры розных каналах пры навучанні і распазнанні і 90% пры аднолькавых. Да таго ж падтрымлівае мала фарматаў запісаў (з папулярных толькі wav з частатой 8 кГц) і мае графічны

інтэрфейс толькі для Windows. Сістэма VoiceCompare расейскай кампаніі «ПронтоТелеком», калі верыць апісанню на сайце [17], гарантуе дакладнасць ужо 93-95%, пры гэтым прызначана толькі для Windows і працуе толькі з тым жа васьмікілагерцовым wav.

На беларускім рынку сістэмы галасавой ідэнтыфікацыі пакуль не распаўсюджаны, таму беларускія прадпрыемствы або карыстаюцца дарагімі расійскімі і яшчэ больш дарагімі амерыканскімі камерцыйнымі прадуктамі, або – часцей за ўсё – увогуле не вырашаюцца ўкараняць такія сістэмы, хоць і маюць у гэтым патрэбу. Таму распрацоўка надзейнай сістэмы ідэнтыфікацыі дыктараў для нашага рынка з'яўляецца задачай актуальнай, хоць і патрабуе добрых алгарытмічных распрацовак, сумесных намаганняў спецыялістаў па апрацоўцы гуку, праграмістаў і добрых маркетолагаў.

2.7. Аналіз існуючых сэрвісаў апрацоўкі маўлення

Большасць існуючых сэрвісаў, якія працуюць з прамовай, такіх, як Google Cloud Speech API, Bing Speech API, API.AI, SpeechKit Cloud API, арыентаваны ў першую чаргу на распазнанне маўлення, у другую чаргу – на сінтэз маўлення, і толькі некаторыя прапануюць дадатковыя функцыі (аналіз намераў у API.AI або сэнсавы разбор у SpeechKit Cloud API). Сервіс 3і Speech Recognition API, які знаходзіцца на момант напісання гэтай працы ў тэставай версіі, прапануе таксама падзел дыялогаў па дыктарам (дыярызацыю), вызначэнне полу, шумоў і паўз. Але бясплатнае карыстанне ім абмежавана тэставым перыядам (як і карыстанне SpeechKit Cloud, і шматлікімі іншымі).

Беларуская анлайн-платформа апрацоўкі прамовы corpus.by [18] вылучаецца вялікай разнастайнасцю функцыяналу: каля пяцідзесяці сэрвісаў, кожны з якіх вырашае асобную задачу накшталт генерацыі транскрыпцыі, падзелу слоў на склады, падліку частотнасці слоў і таму падобнага. Унікальная яна і тым, што акрамя рускай і ангельскай мовы падтрымлівае яшчэ і беларускую – прычым як асноўную мову; і тым, што знаходзіцца ў бясплатным адкрытым доступе. Прадстаўленыя сервісы можна выкарыстоўваць для аўтаматычнай апрацоўкі дадзеных пры стварэнні ўласных лінгвістычных мадэляў. Да таго ж па той прычыне, што сэрвіс распазнання дыктараў там не прапаноўваецца, можна паспрабаваць у якасці такога сэрвісу укараніць распрацаваную сістэму галасавой ідэнтыфікацыі.

2.8. Аналіз залежнасці існуючых сістэм ідэнтыфікацыі дыктара ад мовы

Большасць існуючых сістэм ідэнтыфікацыі дыктара не выкарыстоўваюць асаблівасці мовы, на якой гавораць дыктары. Гэта тычыцца нават сістэм ідэнтыфікацыі дыктараў, якія размаўляюць на некалькіх мовах, такіх, як [21], [22], [23], [24]. Замест гэтага сістэмы распазнання шматмоўных дыктараў

імкнуцца падабраць такую камбінацыю прыкмет, якая менш залежыць ад асаблівасцяў мовы, ахвяруючы пры гэтым якасцю распазнавання для любой асобна ўзятай мовы. Напрыклад, у [23] для гэтага выкарыстоўваюцца прасадычныя прыкметы (інтанацыя і энергія).

Разам з тым сістэмы, дзе распазнаванне заснавана на разбіцці на фанемы, ускосна залежаць ад мовы ўжо таму, што ад яе залежыць набор фанем. Так, у [24] для кожнай мовы выкарыстоўваецца свой «распазнавальнік фанем» (англ. Phone recognizer), і пры распазнаванні маўленне аналізуецца кожным з гэтых «распазнавальнікаў».

Магчымасць выкарыстоўваць асаблівасці мовы існуе як для аднамоўных сістэм, так і для шматмоўных, якія выкарыстоўваюць асобныя аналізатары для кожнай мовы. Звычайна адрозненне такіх аналізатараў не ў выкарыстаных алгарытмах, а ў дадзеных, на якіх адбываецца іх навучанне. Калі мадыфікаваць і самі алгарытмы, можна дамагчыся больш высокай якасці, але вызначыць напрамкі такіх мадыфікацый уяўляецца досыць складанай задачай.

3. Мадэль распрацаванай сістэмы ідэнтыфікацыі дыктара

3.1. Выбар прыкмет для аналізу пры апрацоўцы сігналаў

Пры вырашэнні задачы 3.а фізіялагічныя характарыстыкі прамовы былі выкарыстаны як асноўныя, але пры гэтым было даследавана і ўключэнне ў сістэму дадатковых часавых, статыстычных, некаторых прасадычных прыкмет.

У якасці нізкаўзроўневых спектральных каэфіцыентаў пры навучанні распрацаванай сістэмы можна выкарыстоўваць як MFCC, так і PLP каэфіцыенты (і іх вытворныя). Але эксперыменты паказалі, што PLP каэфіцыенты не далі пераваг у дакладнасці ідэнтыфікацыі ў параўнанні з MFCC, таму было вырашана па змаўчанні выкарыстоўваць мел-частотныя кепстральныя каэфіцыенты як пры навучанні, так і пры распазнаванні дыктараў. Рэалізацыя алгарытму іх вылічэння прадстаўлена ў лістынгу А.4 прыкладання А. Для даследавання магчымасці павелічэння дакладнасці распазнання, звязанай з наяўнасцю ўнікальных характарыстык маўлення чалавека ў галіне высокіх частот, была дададзена опцыя дапаўнення вектара мел-частотных каэфіцыентаў значэннямі каэфіцыентаў IMFCC.

Пры здабыванні прыкмет для кожнага акна сігналу магчыма вылічэнне адвольнай колькасці кароткачасовых прыкмет, якія аб'ядноўваюцца ў адзіны супервектар. Гэта робіцца для кожнай фанемы дыктара наступным чынам:

1) задаюцца вектар сэмплаў сігналу data, аконная функцыя, памер акна ў секундах (windowSize), памер кроку ўзяцця акна (hop), пачатак start і канец end фанемы ў секундах;
2) у залежнасці ад частаты дыскрэтызацыі запісу samplingRate вылічваецца колькасць сэмплаў у акне як windowSize * samplingRate, колькасць сэмплаў у кроку ўзяцця акна як hop * samplingRate і велічыня offset водступаў ад пачатку і канца фанемы;
3) вылічваюцца індэксы пачатку сэмплаў фанемы як max(start * samplingRate – offset, 0) і канца сэмплаў фанемы як min(end * samplingRate + offset, |data|);
4) кожны вектар сэмплаў фанемы (акно) з вылічанным памерам бярэцца з вылічанным крокам і дамнажаецца на вектар аконнай функцыі;
5) для атрыманнага вектару будуецца вектар кароткачасовых прыкмет;
6) вектар дадаецца да супервектару дадзенай фанемы дадзенага дыктара.

Рэалізацыя гэтага алгарытму прадстаўлена ў лістынгу А.2 дадатку А.

Для падаўлення шумоў у якасці перадапрацоўкі сігналу пры навучанні лінгвістычнай мадэлі выкарыстоўвалася выдаленне пастаяннага амплітуднага зрушэння, дызерынг і павелічэнне амплітуды высокачастотных кампанент

сігналу. Таксама выдаляліся доўгія фрагменты з «цішынёй». Пасля вылічэння кепстральных каэфіцыентаў для кожнага дыктара выконваецца нармалізацыя іх матэматычнага спадзявання і дысперсіі для памяншэння ўплыву фонавых шумоў.

Было вырашана не ўключаць у сістэму такія прасадычных характарыстыкі, як тэмп гаворкі ці інтанацыю, бо яны занадта моцна залежаць ад кантэксту паведамлення і да таго ж лёгка паддаюцца наўмыснаму скажэнню. Быў таксама даследаваны ўплыў уключэння эфектыўнай працягласці гучання (якая гаворыць пра сумарную працягласць паўз у маўленні) на якасць распазнавання і ўстаноўлена, што яно не вядзе да істотнага яго паляпшэння. Тым не менш, распрацаваны алгарытм дазваляе ўключаць у сістэму такія характарыстыкі, і, калі будуць знойдзены тыя з іх, што прывядуць да стабільнага паляпшэння якасці, яны будуць выкарыстоўвацца як дадатковыя прыкметы пры фарміраванні мадэлі дыктара.

3.2. Алгарытмы фарміравання мадэляў, іх параўнання і прыняцця рашэнняў

Па той прычыне, што, з аднаго боку, распрацаваная сістэма ідэнтыфікацыі дыктара павінна быць тэкстанезалежнай, а з другога – выкарыстанне выключна нізкаўзроўневай інфармацыі не забяспечвае належнай якасці распазнавання, пры рашэнні задачы 3.b быў зроблены выбар будаваць высокаўзроўневую тэкстазалежную лінгвістычную мадэль толькі пры першапачатковым навучанні і карыстацца ёй пры даданні ў сістэму новых дыктараў. Лінгвістычная мадэль дазваляе выконваць пофанемнае разбіццё навучальных запісаў дыктара, якое выкарыстоўваецца пры фарміраванні яго індывідуальнай мадэлі вымаўлення розных фанем. Пры распазнанні жа лінгвістычная інфармацыя не ўлічваецца і не выконваецца разбіццё тэставага запісу на фанемы, што дазваляе значна паскорыць гэты працэс.

Такім чынам, распрацаваная сістэма выкарыстоўвае наступныя мадэлі:

- адну лінгвістычную мадэль для ўсіх дыктараў;
- індывідуальныя мадэлі вымаўлення фанем для кожнага дыктара.

3.2.1 Пабудова лінгвістычнай мадэлі

Лінгвістычная мадэль фарміруецца паэтапна: на кожным кроку будуецца больш дасканалая мадэль на аснове папярэдняй.

1) Спачатку вылічваюцца нізкаўзроўневыя прыкметы і выконваецца нармалізацыя іх матэматычнага спадзявання і дысперсіі;

2) На аснове знойдзеных нізкаўзроўневых прыкмет і іх вытворных на аснове алгарытмаў мадэлявання гаусавых сумесяў і схаваных маркаўскіх мадэляў

будуюцца мадэлі фанем без выкарыстання інфармацыі аб іх узаемным размяшчэнні і ўплыву яго на гучанне фанем.

3) На аснове мадэляў манафонаў фармуюцца мадэлі трыфонаў, пры гэтым выкарыстоўваюцца не ўсе магчымыя спалучэнні фанем, бо іх колькасць была б роўная колькасці фанем у кубе, а толькі іх найбольш распаўсюджаная частка, якая фарміруецца з дапамогай фанетычнага дрэва рашэнняў.

4) Далей з дапамогай лінейнага дыскрымінантнага аналізу памяншаецца памернасць прасторы ўваходных дадзеных шляхам пабудовы станаў маркаўскіх мадэляў для зыходных вектараў прыкмет.

5) Потым з дапамогай лінейных пераўтварэнняў прыкмет, якія максімізуюць сярэдняе праўдападабенства, у новай рэдуцыраванай прасторы фармуюцца матрыцы трансфармацый для кожнага дыктара. У некаторых сістэмах [19] гэтыя матрыцы выкарыстоўваюцца як прыкметы для ідэнтыфікацыі дыктара, але ў гэтым выпадку памернасць прасторы прыкмет дыктара атрымліваецца вялікай, а алгарытм ідэнтыфікацыі – працаёмкім.

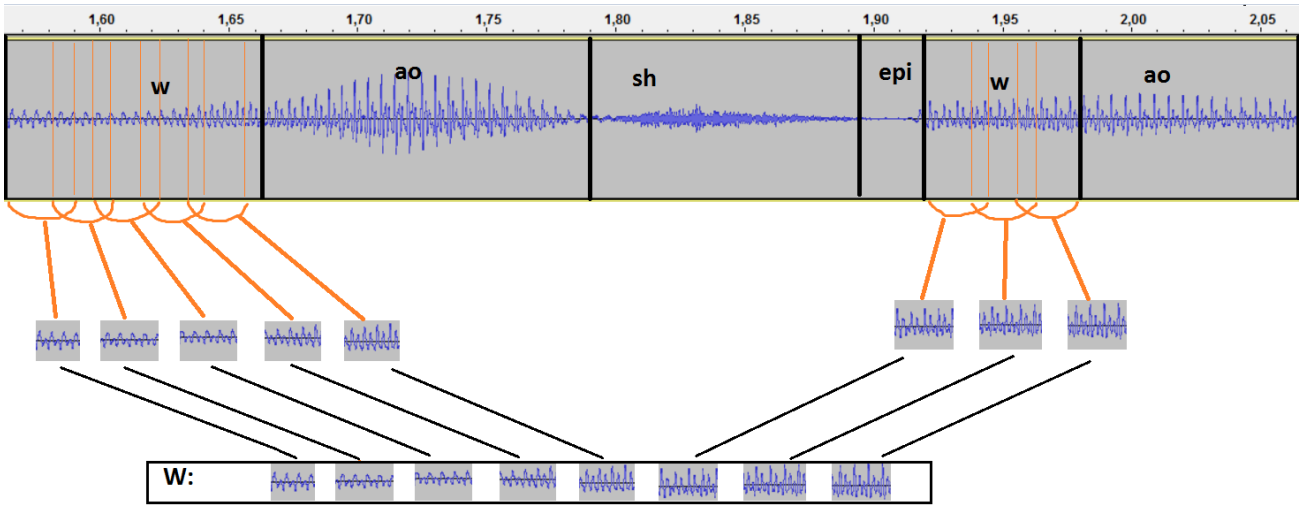
6) На апошнім этапе (англ. SAT, Speaker Adaptive Training) атрыманыя матрыцы трансфармацый выкарыстоўваюцца для нармалізацыі інфармацыі аб дыктарах у мадэлях трыфонаў, дзякуючы чаму разбіццё на фанемы адбываецца ўжо ў прасторы дыктаранезалежных прыкмет. Для выдалення з прыкмет інфармацыі пра дыктара выкарыстоўваецца алгарытм лінейнай рэгрэсіі максімальнага праўдападабенства (англ. FMLLR, Feature Space Maximum Likelihood Linear Regression) [19]: вектары прыкмет запісаў дамнажаюцца на матрыцы, зваротныя матрыцам трансфармацыі іх аўтараў.

7) Для ўдакладнення параметраў пасля навучання кожнай новай мадэлі зноўку выконваецца пафанемное выраўноўванне навучальных запісаў згодна алгарытму навучання Вітэрбі.

Гэтыя крокі апісаны ў скрыпце train_ali.sh (гл. лістынг А.6 прыкладання А).

3.2.2. Пабудова мадэлі дыктара

Пры пабудове мадэлі дыктара вылічаюцца вектары прыкмет для кароткачасовых сумежных адрэзкаў прамовы, што перакрываюцца паміж сабой. Затым яны групуюцца па фанемах, якім належаць, і аб'ядноўваюцца ў супервектары, як паказана на малюнку 3.1. Інфармацыя пра прыналежнасць кожнага адрэзка пэўнай мадэлі дыктара атрымліваецца з дапамогай найбольш дасканалай з лінгвістычных мадэляў, а канкрэтна – SAT-мадэлі.



Малюнак 3.1 – Схема групоўкі адрэзкаў запісу для фанемы

Такім чынам, збіраецца інфармацыя пра вымаўленне дыктарам розных фанем. Для кожнай фанемы будуецца і захоўваецца мадэль гаусіан (гл. лістынг А.1 прыкладання А). Асераднёныя па адрэзках статыстычныя прыкметы, адрозныя ад спектральных каэфіцыентаў і іх вытворных, не ўдзельнічаюць у фарміраванні GMM, а захоўваюцца асобна для забеспячэння аднастайнасці мадэляў гауссовых сумесяў. Дзякуючы таму, што кожная з гэтых прыкмет ўяўляе сабой адзін лік (напрыклад, спектральны цэнтроід або пікавае значэнне сігналу), пры наступным вызначэнні ступені прыналежнасці адрэзка прамовы той ці іншай фанеме можна вылічваць модулі іх рознасці з тымі ці іншымі вагавымі каэфіцыентамі.

Было вырашана не будаваць мадэль дыктара на самым высокім узроўні, прадстаўляючы яго вымаўленне як сумесь вымаўленняў дыктараў рэферэнтнай базы, бо, па-першае, фарміраванне такой базы можа апынуцца вельмі працаёмкім, а па-другое, для яе рэалізацыі пры распазнаванні трэба будаваць паўнавартасную мадэль меркаванага дыктара, каб мець магчымасць пабудаваць яе раскладанне па ўзорных мадэлях. Гэта вядзе да павелічэння патрабаванняў да колькасці і якасці запісаў для распазнавання і павелічэнню вылічальных выдаткаў, у нашым выпадку звязаных з разбіццём запісу на фанемы.

3.2.3. Параўнанне мадэляў і прыняцце рашэнняў

Пры вызначэнні аўтара запісу гэты запіс разбіваецца на адрэзкі аднолькавай даўжыні і для кожнага адрэзка прамовы вылічаецца яго падобнасць на фанемы кожнага дыктара.

Мерай блізкасці вектара прыкмет адрэзка прамовы і фанемы з'яўляецца сярэдняя лагарыфмічная верагоднасць мадэлі гаусіан фанемы, якая вылічана для гэтага вэктару. Калі апроч гаусавых мадэляў для фанемы вылічаны якія-небудзь статыстыкі, яны вылічаюцца і для разгляданага ўчастка сігналу, і да выніку

дадаюцца модулі рознасці адпаведных статыстык, памножаныя на вагавыя каэфіцыенты.

Рашэнне прымаецца метадам k бліжэйшых суседзяў: для адрэзка шукаецца k фанем з мінімальнай адлегласцю і выбіраецца дыктар, якому належыць найбольшая колькасць гэтых фанем. Магчымы і ўзважаны варыянт, калі выбіраецца дыктар сярод аўтараў k бліжэйшых фанем, фанемы якога ў сярэднім найбольш блізкія да разглядаемага адрэзку прамовы.

Алгарытм вызначэння дыктару па кароткачасовых прыкметах пры k=1 выглядае наступным чынам:

1) задаюцца вектар сэмплаў сігналу data, аконная функцыя, памер акна ў секундах (windowSize), памер кроку ўзяцця акна (hop);
2) у залежнасці ад частаты дыскрэтызацыі запісу samplingRate вылічваецца колькасць сэмплаў у акне як windowSize * samplingRate і колькасць сэмплаў у кроку ўзяцця акна як hop * samplingRate;
3) кожны вектар сэмплаў запісу (акно) з вылічанным памерам бярэцца з вылічанным крокам, далей кожнае такое акно:
   a) дамнажаецца на вектар аконнай функцыі;
   b) для атрыманнага вектару будуецца вектар кароткачасовых прыкмет;
   c) для кожнай фанемы кожнага дыктара вылічваецца верагоднасць яе падабенства вектару прыкмет;
   d) для дыктара, якому належыць фанема з самай вялікай верагоднасцю, павялічваецца лічыльнік;
4) дыктар з найбольшым значэннем лічыльніка выбіраецца як самы верагодны аўтар запісу.

Рэалізаваны алгарытм прыняцця рашэння прадстаўлены ў лістынгу А.3 прыкладання А.

Пры прыняцці рашэння на адкрытым мноcтве дыктараў улічваецца верагоднасць аўтарства ў параўнанні з астатнімі дыктарамі. Калі яна нізкая, дыктар разглядаецца як невядомы.

3.3. Архітэктура сістэмы

Распрацаваная праграма ўключае ў сябе пяць падсістэм:

- падсістэма здабывання прыкмет;
- падсістэма фарміравання лінгвістычнай мадэлі;
- падсістэма фарміравання мадэляў дыктараў;
- падсістэма распазнавання дыктараў;
- падсістэма карыстальніцкага інтэрфейсу.

Кампаненты гэтых падсістэм прадстаўлены ў табліцы 3.1; там жа апісаны задачы, якія выконваюцца кожным модулем.

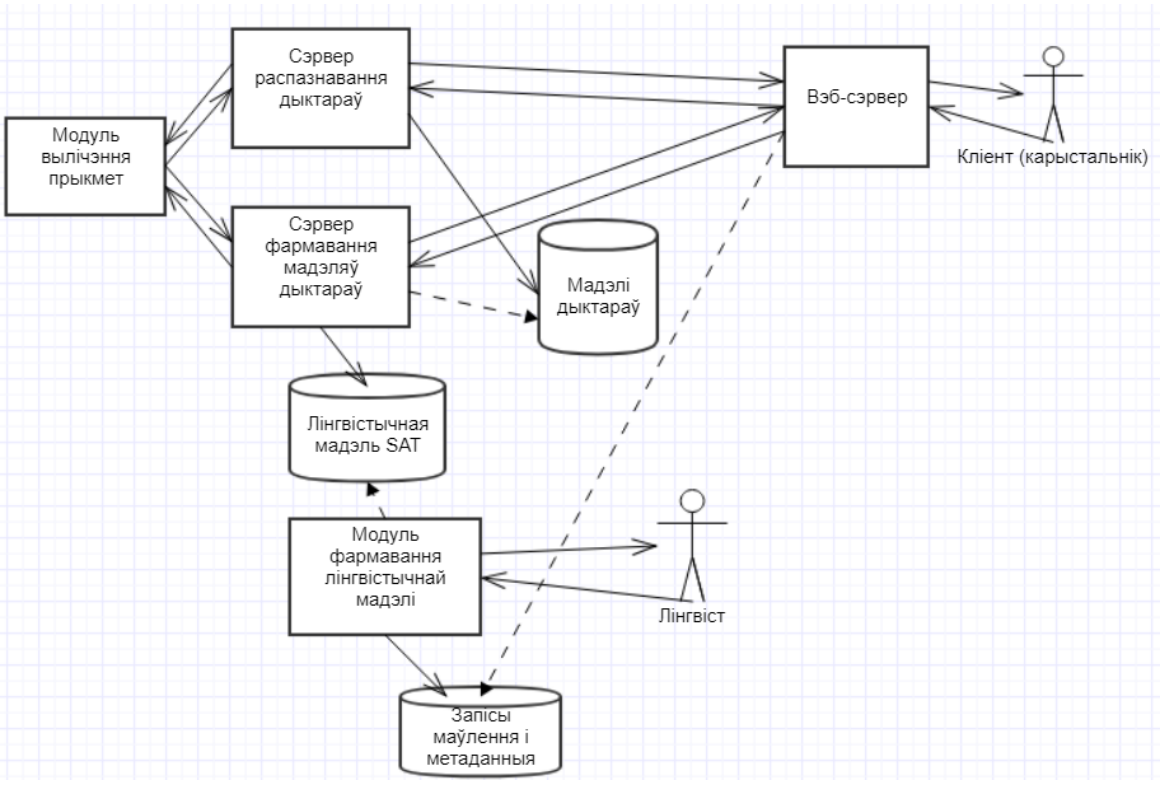
Табліца 3.1 – Апісанне модуляў распрацаванай сістэмы ідэнтыфікацыі дыктара

Падсістэма	**Модуль падсістэмы**	**Задача модуля**
Знаходжанне прыкмет	Чытанне запісаў	Счытванне гукавых файлаў і прывядзенне іх да адзінага фармату
	Папярэдняя апрацоўка сігналаў	Апрацоўка дадзеных з мэтай падаўлення шумоў
	Разбіццё на вокны	Разбіццё запісаў на кароткачасовыя адрэзкі і прымяненне да іх аконнай функцыі
	Вылічэнне прыкмет	Вылучэнне прыкмет ў гукавым сігнале
Пабудова лінгвістычнай мадэлі	Мадэляванне манафонаў	Фарміраванне акустычных мадэляў дадзеных пры навучанні, незалежных ад узаемнага размяшчэння элементаў (фанем)
	Мадэляванне LDA-MLLT трыфонаў	Фарміраванне кантэкстна-залежных акустычных мадэляў дадзеных пры навучанні на аснове алгарытмаў лінейнага дыскрымінантнага аналізу і лінейных пераўтварэнняў прыкмет, якія максімізуюць сярэдняе праўдападабенства (англ. Linear Discriminant Analysis – Maximum Likelihood Linear Transform)
	Мадэляванне SAT трыфонаў	Фарміраванне кантэкстна-залежных акустычных мадэляў дадзеных пры навучанні з выкарыстаннем нармалізацыі дыктараў (англ. Speaker Adaptive Training)
Пабудова мадэляў дыктараў	Выраўноўванне межаў фанем	Пафанемнае выраўноўванне запісаў пры навучанні
	Пабудова мадэляў фанем	Пабудова GMM-мадэляў фанем пры навучанні
Распазнаванне дыктараў	Параўнанне з мадэлямі фанем	Вылічэнне ступені падабенства кароткіх неўпарадкаваных адрэзкаў запісу мадэлям фанем
	Вылічэнне дадатковых прыкмет	Вылічэнне неспектральных (напрыклад, статыстычных) прыкмет для кароткіх неўпарадкаваных адрэзкаў запісу

Працяг табліцы 3.1

Падсістэма	**Модуль падсістэмы**	**Задача модуля**
Распазнаванне дыктараў	Вызначэнне дыктара	Прыняцце рашэння наконт прыналежнасці запісу таму ці іншаму дыктару
Карыстальніцкі інтэрфейс	Вэб-сервіс (узаемадзеянне з карыстальнікамі)	Даванне карыстальнікам інтэрфейсу да падсістэм фарміравання мадэляў дыктараў і распазнавання запісаў
	Скрыпты (узаемадзеянне з ўнутранымі карыстальнікамі)	Даванне карыстальнікам-лінгвістам інтэрфейсу для паляпшэння лінгвістычнай мадэлі шляхам дабаўлення запісаў для навучання і папаўнення лексікона

Сістэма прадстаўлена вэб-серверам, серверам для фарміравання мадэляў дыктараў, серверам для распазнання дыктараў па запісах іх прамовы, модулем здабывання прыкмет і модулем для фарміравання лінгвістычнай мадэлі, як паказана на малюнку 3.2.



Малюнак 3.2 – Архітэктура сістэмы ідэнтыфікацыі дыктара

Вэб-сервіс дае інтэрфейс да падсістэм фарміравання мадэляў дыктараў і распазнавання гэтых дыктараў па іх запісах, а доступ да падсістэмы фарміравання лінгвістычнай мадэлі прадастаўляецца толькі унутраным карыстальнікам сістэмы (лінгвістам) праз камандны радок.

Першапачаткова лінгвістамі фармуюцца базы запісаў маўлення для выбраных моў (беларускай і ангельскай), якія выкарыстоўваюцца пры фарміраванні лінгвістычных мадэляў для гэтых моў. Далей гэтыя мадэлі выкарыстоўваюцца серверам, які стварае мадэлі дыктараў на аснове іх запісаў, атрыманых з вэб-сервера (таксама гэта могуць быць загадзя нарыхтаваныя запісы, якія паказваюцца карыстальнікам у якасці прыкладу пры дэманстрацыі працы сістэмы). Мадэлі дыктараў выкарыстоўваюцца падсістэмай, якая адказвае за іх распазнаванне.

Запісы, атрыманыя ад карыстальнікаў пры распазнаванні, захоўваюцца на серверы толькі ў межах запыту. Запісы, атрыманыя ад карыстальнікаў пры даданні імі дыктараў у сістэму, назапашваюцца на вэб-серверы і перыядычна выкарыстоўваюцца для паляпшэння лінгвістычнай мадэлі шляхам яе паўторнай пабудовы. Пры гэтым ёсць некалькі варыянтаў фарміравання лексікону:

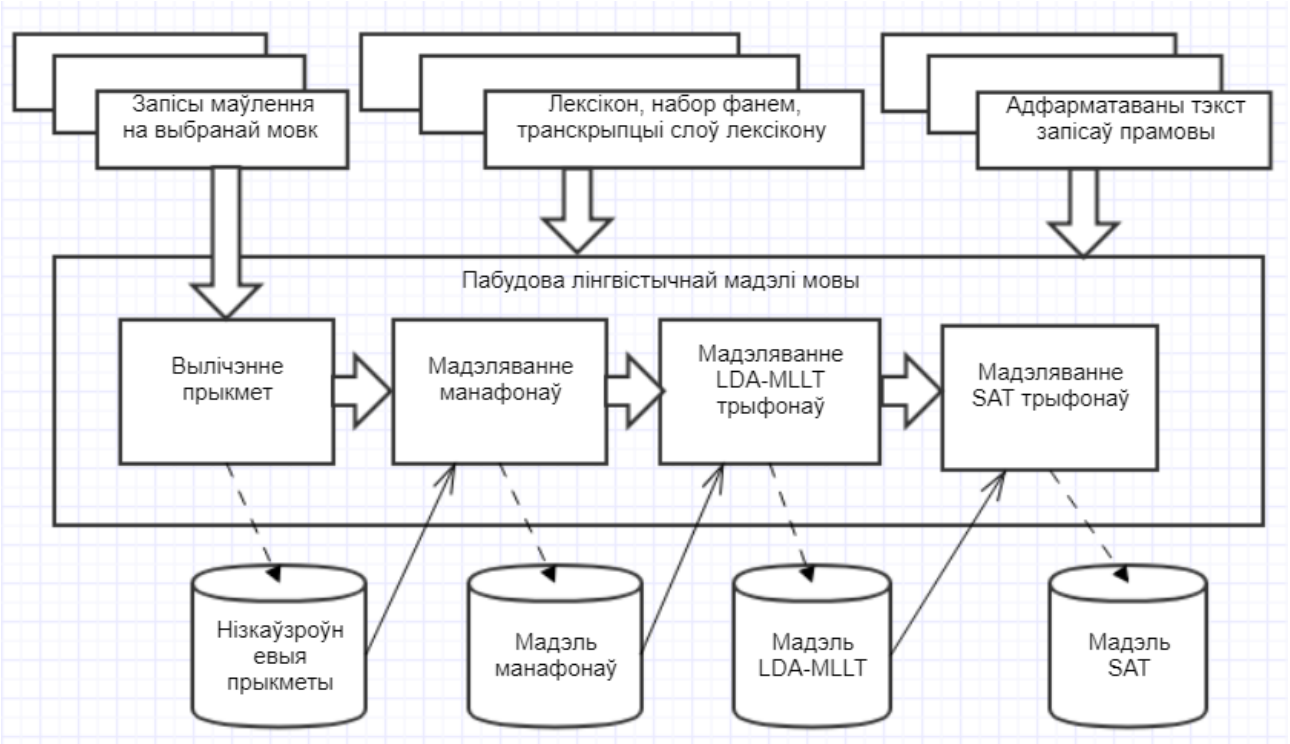
1) выкарыстоўваць першапачаткова створаны для дадзенай мовы лексікон;
2) даць карыстальнікам магчымасць папаўняць лексікон;
3) даць унутраным карыстальнікам-лінгвістам магчымасць папаўняць лексікон.

У выпадку давання карыстальнікам магчымасці дадаваць словы ў лексікон неабходны ўдзел лінгвіста для ацэнкі таго, ці сапраўды прапанаваныя словы існуюць ці пачынаюць існаваць у мове.

3.4. Асноўныя сцэнары выкарыстання

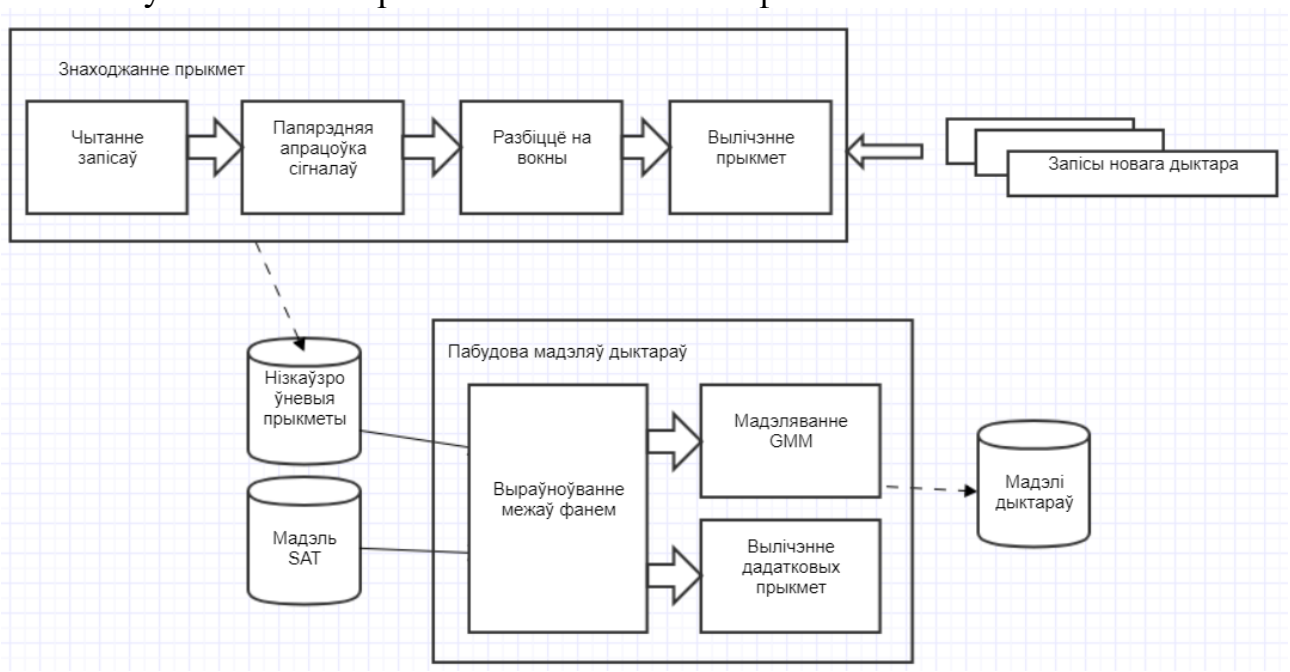
Праца сістэмы ідэнтыфікацыі дыктара працякае па адным з трох асноўных сцэнарыяў:

- спецыяліст фармуе лінгвістычную мадэль для абранай мовы;
- карыстальнік дадае новага дыктара ў сістэму для абранага датасэта;
- карыстальнік распазнае аўтара запісу сярод дыктараў абранага датасэта.



Малюнак 3.3 – Схема фарміравання лінгвістычнай мадэлі

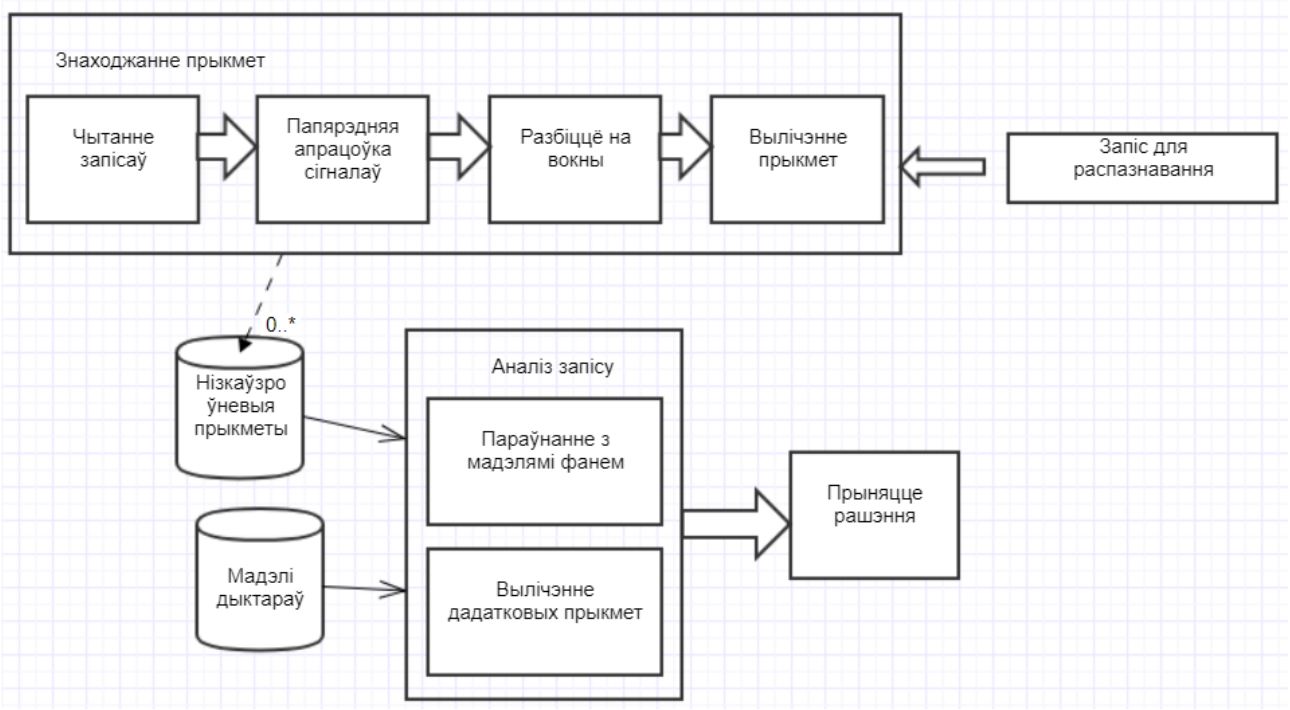
Схема фарміравання лінгвістычнай мадэлі мовы, якая дае магчымасць выконваць разбіццё маўлення на фанемы, прадстаўлена на малюнку 3.3. Ад якасці гэтай мадэлі будзе моцна залежаць дакладнасць распазнавання, таму пажадана, каб сумарная працягласць ўваходных запісаў была вялікая і каб іх змест быў максімальна разнастайны лексічна і фанетычна.



Малюнак 3.4 – Схема дадання новага дыктара ў сістэму ідэнтыфікацыі

Схема дадання новага дыктара ў сістэму прадстаўлена на малюнку 3.4. Запісы апрацоўваюцца незалежна ад запісаў астатніх дыктараў, апрацоўка ўключае вылічэнне нізкаўзроўневых характарыстык маўлення і фарміраванне

гаусавых мадэляў вымаўлення ім фанем на аснове гэтых характарыстык. Каб пазбавіць карыстальніка ад неабходнасці суправаджаць свае запісы граматна напісаным тэкстам і забяспечыць паўнату сфармаваных мадэляў, можна прапаноўваць яму чытаць нарыхтаваныя абзацы, што складаюцца так, каб ўключаць у сябе максімальную колькасць фанем мовы.



Малюнак 3.5 – Схема распазнання дыктара па запісе яго маўлення

Схема распазнання аўтара запісу маўлення (рашэння задачы 3.с) прадстаўлена на малюнку 3.5. Магчымы як непасрэдны запіс з мікрафона, так і загрузка існуючага аўдыёфайла аднаго з фарматаў, якія падтрымліваюцца сістэмай. Патрабуецца толькі каб гаворка была на мове абранага датасэта.

4. Рэалізацыя сістэмы ідэнтыфікацыі дыктара

4.1. Стварэнне акустычнай базы дадзеных для беларускай мовы

Калі для ангельскай мовы існуюць гатовыя акустычныя базы, то для беларускай мовы стварэнне такой базы – асобная і досыць працаёмкая задача. У АІПІ НАН Беларусі за гэтую задачу ўзяліся, было вырашана ствараць акустычную базу на аснове чытання дыктарамі нарыхтаванага тэксту. На момант, калі аўтар далучыўся да каманды, да начыткі тэкстаў было прыцягнута каля шасцідзесяці носьбітаў мовы, а таксама быў складзены слоўнік праграмна згенераваных і праверaных лінгвістамі фанетычных транскрыпцыі. Стварэнне акустычнай базы выбраным спосабам патрабуе вырашэння наступных задач:

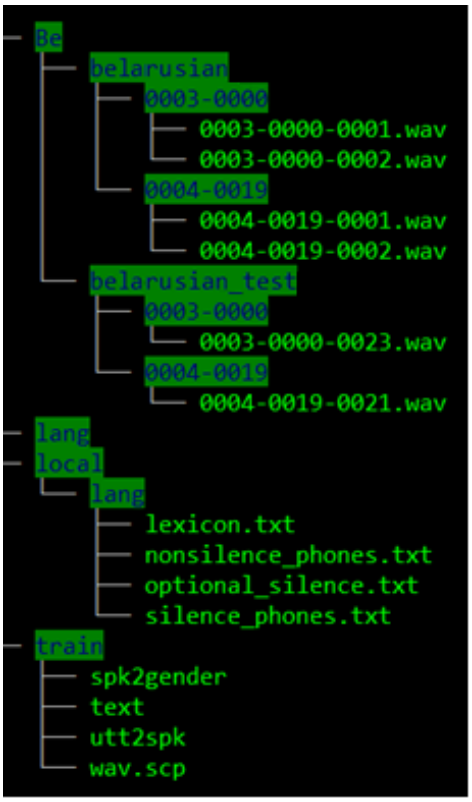
1) падрыхтоўка тэкставага корпуса;
2) падрыхтоўка запісаў маўленчага сігналу, адпаведнага тэкставаму корпусу;
3) структурызацыя атрыманых дадзеных і суправаджэнне іх метададзенымі.

Перад начыткай прыгатаваны загадзя тэкставы корпус на беларускай мове разбіваецца на пранумераваныя невялікія абзацы па некалькі сказаў. Запісы захоўваюцца ў фармаце wav, называюцца па правілу «нумар дыктара» + «-» + «нумар тэксту» + «-» + «нумар абзаца» + «.wav» і згрупаваны па папках з імёнамі ў фармаце «нумар дыктара» + «-» + «нумар тэксту». Для стварэння акустычнай базы на аснове наяўных запісаў прамовы і тэксту адпаведных абзацаў, гэта значыць для вырашэння апошняй задачы, выкарыстоўваецца наступны алгарытм:

1) стварыць «вычышчаны» тэкст на аснове наяўнага, для чаго
    a. выдаліць знакі прыпынку;
    b. замяніць абрэвіятуры, скарачэння, колькасці, сімвалы, што чытаюцца ўслых, і т.п. словамі ў той форме, у якой яны чытаюцца дыктарам (але пры магчымасці лепш рабіць гэта да начыткі, пакідаючы спосаб расшыфроўкі за лінгвістам, а не за дыктарам);
2) адправіць атрыманы тэкст сэрвісу TranscriptionGenerator [20], выкарыстоўваючы яго праграмны інтэрфейс, і сфарміраваць на аснове яго адказу файл з лексіконам і транскрыпцыяй для слоў, якія сустракаюцца ў абзацы;
3) калі ўсе запісы будуць апрацаваны, сфармаваць агульны лексікон, аб'яднаўшы файлы для наяўных урыўкаў, адсартаваўшы іх і выдаліўшы радкі, якія паўтараюцца.

Атрыманая база мае структуру, паказаную на малюнку 4.1. Частка запісаў не апрацоўваецца, вынесена ў папку belorussian_test і можа выкарыстоўвацца пры тэставанні сістэмы – напрыклад, пры крос-валідацыі. Для пабудовы лінгвістычнай мадэлі, апроч здымкаў маўлення, патрабуецца іх тэкставая

транскрыпцыя, файл з выкарыставанымі фанемамі, файлы з фанемамі, якія пазначаюць цішыню і нераспазнаныя фанемы, файл з лексіконам, дзе на кожным радку размяшчаецца слова і яго фанетычная транскрыпцыя, а таксама файлы з адпаведнасцю запісаў дыктару, полам дыктараў і шляхамі да навучальных запісаў. Структура гэтых файлаў указана ў дадатку В.



Малюнак 4.1 – Структура акустычнай базы дадзеных

Атрыманая база змяшчае дзве тысячы чатырыста шаснаццаць запісаў пяцідзесяці васьмі дыктараў, сумарная працягласць якіх, палічаная з дапамогай праграмы SoXI, складае 11.8 гадзін. Пасля дадання начытаных аўтарам фрагментаў тэксту працягласцю чатыры хвіліны ў базе стала дзве тысячы чатырыста трыццаць пяць запісаў пяцідзесяці дзевяці дыктараў.

4.2. Стварэнне акустычных баз дадзеных для ангельскай мовы

Для стварэння акустычных баз дадзеных для ангельскай мовы было абрана два маўленчых корпуса: TIMIT – класічны корпус запісаў прамовы амерыканскіх дыктараў, прызначаны для ацэнкі работы сістэм распазнання прамовы, і адкрыты бясплатны корпус LibriSpeech, сабраны з запісаў чытання дыктарамі аўдыёкніг.

Корпус TIMIT змяшчае па дзесяць запісаў шасцісот трыццаці дыктараў; працягласць кожнага запісу складае каля двух секунд. Сумарная працягласць запісаў, палічыная з дапамогай SoXI, склала 4.7 гадзін (але на сайце TIMIT паказана 5.4 гадзіны).

Выбарка з корпуса LibriSpeech змяшчае пяць тысяч трыста дваццаць тры запісы васьмідзесяці дыктараў працягласцю каля трох секунд кожная. Сумарная працягласць запісаў, палічыная з дапамогай SoXI, склала 10.8 гадзін.

Структуру корпуса LibriSpeech давялося крыху змяніць, бо ў ёй у тэставай і навучальнай выбарках знаходзіліся запісы розных дыктараў. Гэта нармальна пры рашэнні задачы пераўтварэння маўлення ў тэкст, але для задачы ідэнтыфікацыі дыктара патрэбен аднолькавы набор дыктараў для навучання і распазнавання.

Таксама былі зменены файлы адпаведнасці спікераў запісам. У LibriSpeech запісы аднаго і таго ж дыктара ў розных аўдыторыях лічыліся запісамі розных дыктараў. Гэта было зроблена для павелічэння ступені нармалізацыі дыктараў у выпадку выкарыстання алгарытму лінейнага пераўтварэння прыкмет, які максімізуе сярэдняе праўдападабенства, пры фарміраванні лінгвістычнай мадэлі. Але для задачы ідэнтыфікацыі дыктара вынікі апынуліся лепш пры выкарыстоўванні сапраўдных суадносін.

4.3. Выбар інструментаў для распрацоўкі і разгортвання праграмы

Для распрацоўкі сэрвісаў, якія выконваюць мадэляванне і распазнаванне запісаў, была абрана мова праграмавання C++ у сілу яе кросплатформавасці, высокай хуткасці вылічэнняў, адносна лёгкай пераноснасці кода на C++ на мабільныя платформы і ўбудаваныя сістэмы і ў той жа час адноснай высокаўзроўневасці, а таксама ў сілу наяўнасці вялікай колькасці праверaных рэалізацый алгарытмаў апрацоўкі сігналаў і імавернаснага мадэлявання.

Пры выбары бібліятэк з адкрытым зыходным кодам, якія можна выкарыстоўваць для аналізу маўлення з мэтай выдзялення фанем, улічваліся вынікі, атрыманыя ў [27]. Сярод такіх кандыдатаў, як Hidden Markov Model Toolkit, HDecode і Julius, Pocketsphinx і Sphinx-4, Kaldi [28], апошняя з'яўляецца найбольш перадавым на дадзены момант інструментарыем працы з маўленнем, які паказаў найлепшыя вынікі пры вырашэнні задачы пераўтварэння вуснага маўлення ў тэкст. Для распрацаванай сістэмы ідэнтыфікацыі дыктара не трэба атрыманне выніковай тэкставай транскрыпцыі запісу, але трэба атрыманне разбіцця яе на фанемы. Таму было вырашана выкарыстоўваць Kaldi пры фарміраванні лінгвістычнай мадэлі.

Для напісання скрыптоў, якія ствараюць тэкставыя файлы з метаінфармацыяй для Kaldi і сэрвісу пабудовы мадэляў, выкарыстоўваліся скрыптовыя мовы Python і Perl; для напісання скрыптоў, якія запускалі праграмы Kaldi з патрэбнымі параметрамі, выкарыстоўваўся bash. У гэтым плане падтрымліваўся стыль працы з дадзенымі, прыняты ў Kaldi.

Для атрымання статыстычных, часавых, спектральных, танальных, рытмічных і іншых характарыстык сігналу была абрана кросплатформавая open-source бібліятэка Essentia. У параўнанні з аналагамі (Marsyas, OpenSmile) яна прапаноўвае найбольшую разнастайнасць даступных прыкмет маўлення, мае добрую дакументацыю і зручна спраектаваныя мадэлі вылічэння. Пры стварэнні мадэляў гаусавых сумесяў выкарыстоўвалася таксама кросплатформавая open-source бібліятэка лінейнай алгебры і матрычных вылічэнняў Armadillo.

Для чытання ўваходных дадзеных пры распазнаванні дыктара і пераўтварэнні іх да фармату PCM, акрамя стандартнай бібліятэкі Libsndfile, выкарыстана кросплатформавая бібліятэка Mpg123, бо Libsndfile не падтрымлівае фармат MPEG з-за патэнтавых абмежаванняў, якія існавалі на момант яе стварэння. Дзякуючы гэтаму падтрымліваецца мноства фарматаў ўваходных файлаў пры навучанні (wav, sphere, mp3, flac, speex і іншыя). Рэсэмпліраванне, мікшыраванне і выдаленне цішыні робіцца з дапамогай уціліты sox.

Для распрацоўкі вэб-сэрвісу, які прадастаўляе графічны інтэрфейс, абрана мова Java дзякуючы яе, зноў жа, кросплатформавасці і наяўнасці фрэймворкаў, якія робяць зручным стварэнне REST вэб-праграм (у прыватнасці, для распрацаванай праграмы выкарыстоўваўся Spring). Для стварэння вэб-старонак выкарыстоўваўся стэк HTML, CSS, JavaScript, Bootstrap, JQuery.

Разгортванне выконваецца з дапамогай платформы Docker і сістэм аўтаматычнай зборкі CMake (для C++ сэрвісаў) і Gradle (для вэб-сэрвісу). Кожны сэрвіс запускаецца ў асобным докер-кантэйнеры, абмен дадзеных паміж сэрвісамі ажыццяўляецца праз сокеты і сумесныя папкі. Канфігурацыя мапінга партоў HTTP, сокетаў і агульных папак апісваецца ў файле docker-compose, які дазваляе запускаць сістэму адной камандай. Вэб-сэрвіс запускаецца на ўбудаваным сэрвэры Apache Tomcat.

4.4. Рэалізацыя асноўных модуляў праграмы

Падсістэма здабывання прыкмет, выкарыставаная для вырашэння задачы 3.а, прадстаўлена дынамічнай бібліятэкай FeatureExtractor, якая ўключае класы SoundFile (счытванне файлаў і прывядзенне да адзінага фармату), FeatureExtractor (абстрактны клас здабывання прыкмет), MfccExtractor (вылічэнне MFCC), Preprocessor (папярэдняя апрацоўка сігналу), падпраграмы для разбіцця на вокны. Таксама падтрымліваецца магчымасць выкарыстоўваць пры навучанні мадэляў дыктара прыкметы, сфармаваныя на этапе пабудовы лінгвістычнай мадэлі, таму FeatureExtractor уключае падпраграмы для счытвання гэтых прыкмет і пераўтварэння з унутранага фармату Kaldi ў фармат, неабходны для далейшай пабудовы мадэляў дыктараў. Але трэба памятаць, што гэтыя прыкметы аптымізаваныя для вырашэння іншай задачы (разбіцця на фанемы), і

для дасягнення большай дакладнасці лепш здабываць прыкметы з дапамогай спецыяльна распрацаваных для гэтага падпраграм дадзенай бібліятэкі.

Для пабудовы лінгвістычнай мадэлі напісаны скрыпт train_ali.sh (гл. лістынг А.6 прыкладання А). Файлы, якія атрымліваюцца пры разбіцці на фанемы з дапамогай гэтай мадэлі, прыводзяцца да ўнутранага фармата з дапамогай perl-скрыптоў. Так, у лістынгу А.5 прыкладання А прыведзены скрыпт fotmat_ali_single.pl, які фармаціруе файлы разбіцця на фанемы запісаў новага дыктара.

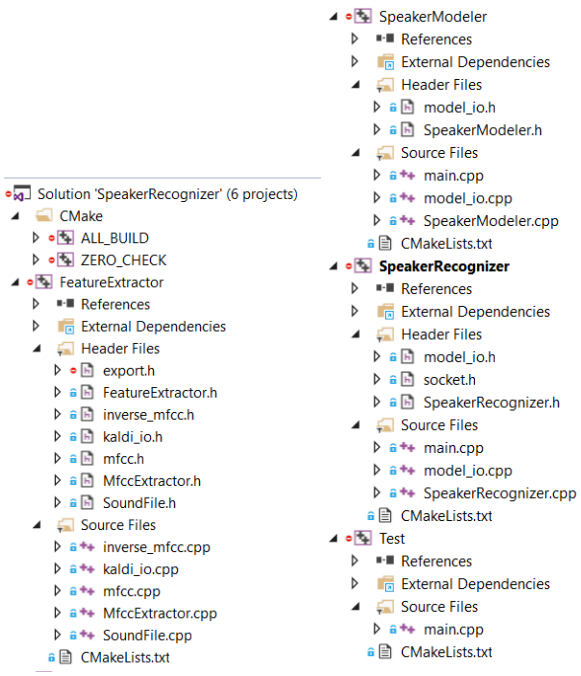
Для выкліку і канфігурацыі неабходных скрыптоў Kaldi і скрыптоў фармаціравання пры разбіцці на фанемы запісаў новага дыктара напісаны скрыпт process.sh, прадстаўлены ў лістынгу А.9. Па ягоным выкананні ствараюцца файлы, дзе пазначаны межы фанем у запісах дыктара, а таксама дапаможны файл з ўнутранымі ідэнтыфікатарамі запісаў, якія выкарыстоўваюцца ў далейшым пры навучанні мадэляў дыктара.

Падсістэма пабудовы мадэляў дыктара прадстаўлена праграмай SpeakerModeler, у выніку выканання якой у названай папцы ствараюцца бінарныя файлы, якія захоўваюць мадэлі фанем кожнага дыктара, а таксама тэкставы файл model.txt, дзе пазначана колькасць дыктараў і адпаведнасць мадэляў фанем дыктарам. Прыклад запуску гэтай праграмы прыведзены ў скрыпце model.sh ў лістынгу А.7 прыкладання А.

Для распазнавання аўтараў запісаў патрабуецца толькі файл са шляхамі да гэтых запісаў. Падсістэмы аналізу распазнавання запісу і прыняцця рашэння рэалізаваны ў праграме SpeakerRecognizer, у выніку выканання якой фарміруецца тэкставы файл з указаннем ідэнтыфікатараў дыктараў для кожнага з названых запісаў. Прыклад запуску гэтай праграмы прыведзены ў скрыпце test.sh у лістынгу А.8 прыкладання А.

Для ацэнкі якасці распазнавання сістэмай дыктараў створана невялікая ўтыліта Test, якая параўноўвае адказы сістэмы з зыходнымі і падлічвае колькасць правільных адказаў.

Структура файлаў C++ сэрвісаў і ўтыліт прадстаўлена на малюнку 4.2.

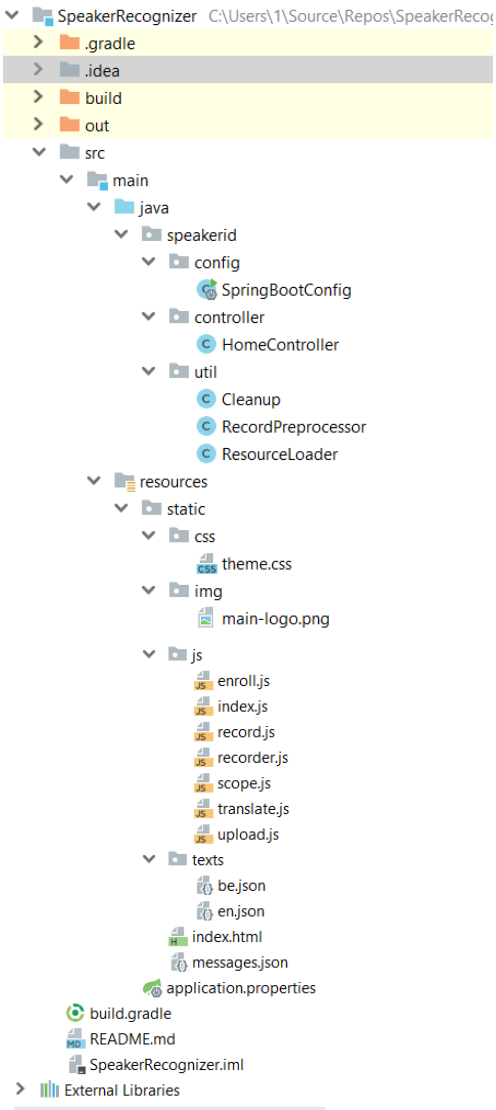


Малюнак 4.2 – Структура файлаў C++ сэрвісаў

Вэб-інтэрфейс уяўляе сабой REST-сэрвіс. Запыты апрацоўваюцца ў кантролеры (клас HomeController), за перыядычную ачыстку файлаў, створаных пры апрацоўцы карыстацкі дадзеных, адказвае клас Cleanup, за загрузку даных з файлавай сістэмы – клас ResourceLoader. Клас SpringBootConfig адказвае за загрузку і канфігурацыю прыкладання, RecordPreprocessor – за папярэднюю апрацоўку карыстальнікіх запісаў, якая ўключае рэсэмпліраванне да 16 кГц, мікшыраванне ў адзін канал і выдаленне цішыні працягласцю больш за палову секунды.

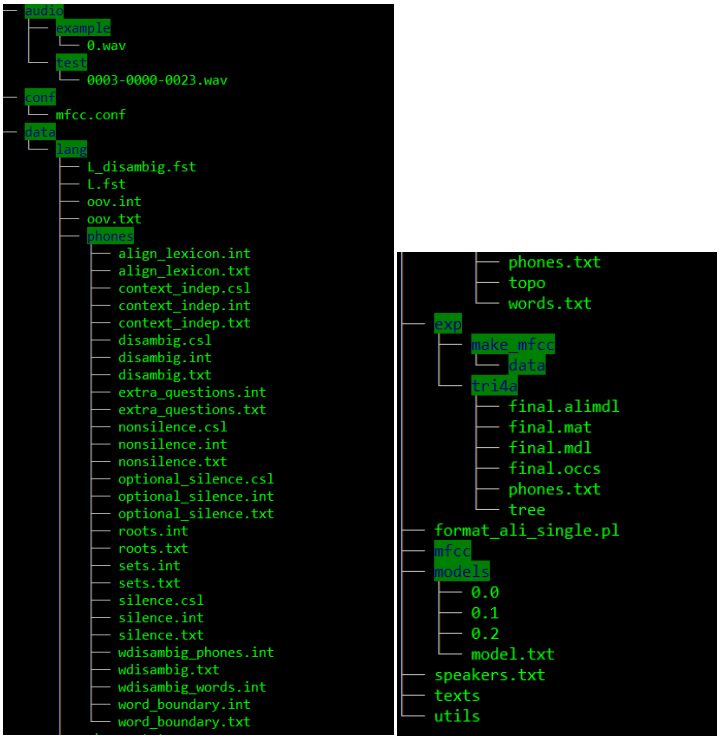
У json файлах be.json і en.json захоўваюцца фрагменты тэкстаў, якія прапануецца прачытаць карыстальніку для дадання свайго голасу ў датасэт. Кожны прачытаны фрагмент правяраецца на адпаведнасць тэксту. Калі ён прачытаны правільна, для яго ствараецца файл з разбіццём на фанемы. Калі для ўсіх запісаў створана разбіццё на фанемы, карыстальнік можа адправіць іх на апрацоўку, пасля якой мадэлі яго фанем будуць улічвацца раўнапраўна з мадэлямі іншых дыктараў абранага датасэта.

Сайт аднастаронкавы, разметка знаходзіцца ў файле index.html, а логіка інтэрфейсу – у javascript-файлах папкі js. Структура файлаў праграмы прадстаўлена на малюнку 4.3.



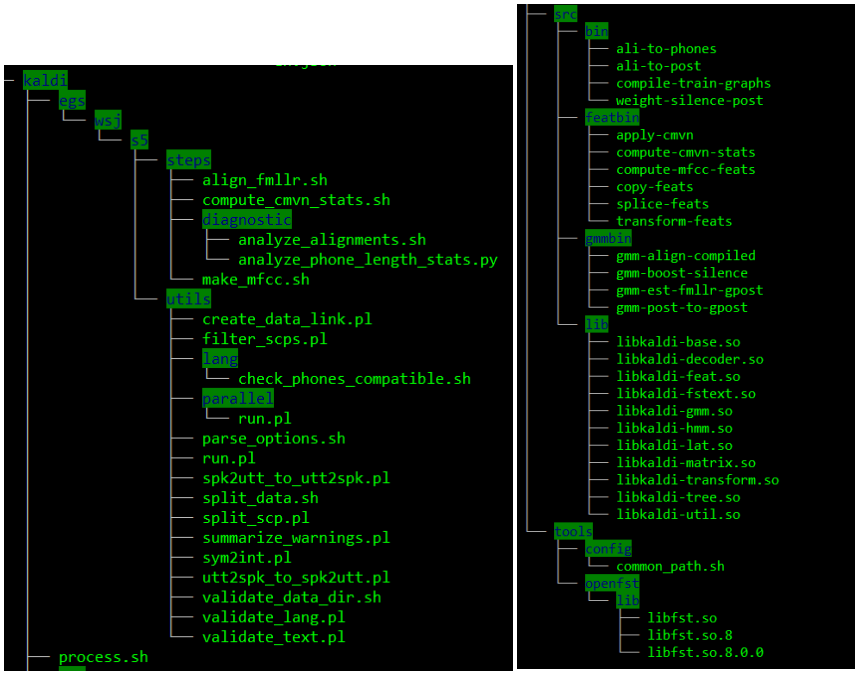
Малюнак 4.3 – Структура файлаў вэб-сэрвісу

Для кожнага датасэта створана папка з запісамі дыктараў, лінгвістычнай мадэллю, інфармацыяй пра мову і дыктараў, дапаможнымі скрыптамі. Яе структура прадстаўлена на малюнку 4.4. Для кожнага карыстальніка ў папках data, exp, mfcc і exp/make_mfcc/data ствараюцца і перыядычна ачышчаюцца адпаведныя гэтаму карыстальніку папкі з яго дадзенымі.



Малюнак 4.4 – Структура файлаў датасэта

На малюнку 4.5 прадстаўлена структура файлаў Kaldi, якія выкарыстоўваюцца для фанемнага выраўноўвання на аснове загадзя загружаных на сервер лінгвістычных мадэляў.



Малюнак 4.5 – Структура файлаў Kaldi

На дадзены момант вэб-інтэрфейс змяшчае панэль выведу і панэль з наступнымі ўкладкамі:

- Example / Прыклад;
- Speakers / Дыктары;
- Record / Запісаць;

- Upload file / Загрузіць файл;
- Add yourself / Дабавіць сябе.

Падтрымліваецца інтэрфейс на англійскай і беларускай мовах з магчымасцю выбару мовы. Змест ўкладак прадстаўлены ў прыкладанні С.

4.5. Магчымасці паралелізацыі вылічэнняў

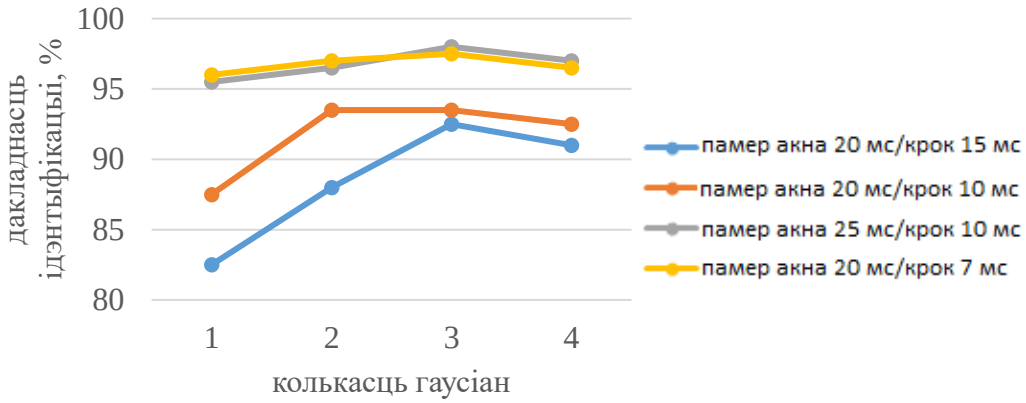
Па той прычыне, што навучанне лінгвістычнай мадэлі і мадэляў дыктараў патрабуе вялікіх часавых затрат, распрацаваная сістэма падтрымлівае шматпаточнасць. Самы працяглы працэс – фарміраванне лінгвістычнай мадэлі – можа выконвацца на некалькіх кластарах адначасова дзякуючы таму, што сістэма падтрымлівае працу з праграмным забеспячэннем для размеркаваных вылічэнняў, такім, як Sun Grid Engine і Slurm Workload Manager; падтрымліваецца і звычайнае шматпаточнае выкананне на адной машыне. Жаданая ступень і від паралелізацыі задаюцца ў адпаведных файлах канфігурацыі.

Праца кліенцкай часткі праграмы распаралельваецца з дапамогай пулаў патокаў, якія ствараюцца ў кожным сэрвісе.

5. Вынікі камп'ютэрных эксперыментаў

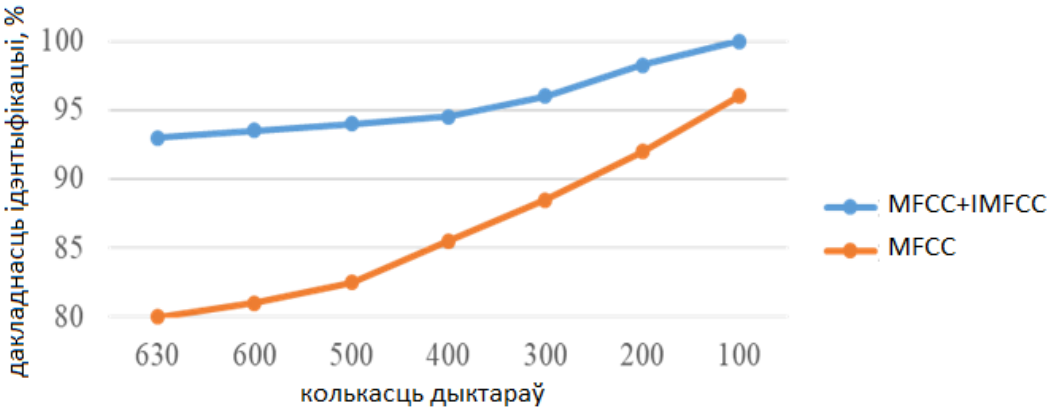
Якасць распазнавання дыктараў ацэньвалася з дапамогай крос-валідацыі, датасэты разбіваліся ў суадносінах 4:1.

Былі праведзены эксперыменты з рознай колькасцю гаусіан пры навучанні GMM фанем дыктара (гл. малюнак 5.1) і ўстаноўлена, што аптымальная колькасць гаусіан раўна тром. Меншай колькасці недастаткова для мадэлявання размеркавання верагоднасці фанемы, а пры большым многія фанемы прапускаюцца пры навучанні, бо маюць занадта малую працягласць. Для таго, каб для фанемы можна было пабудаваць GMM, трэба, каб колькасць участкаў сігналу, якімі гэта фанема прадстаўлена пры разбіцці яго на вокны, была не менш, чым колькасць гаусіан у мадэлі. Так, калі фанема доўжыцца 35 мс і запіс разбіваецца на вокны па 25 мс праз 10 мс, яна будзе прадстаўлена двума вектарамі і не будзе ўлічаная, калі колькасць гаусіан будзе больш за дзве.



Малюнак 5.1 – Залежнасць дакладнасці ідэнтыфікацыі ад памеру і кроку вокнаў і колькасці гаусіан (для ста дыктараў корпуса TIMIT)

Пры тэставанні распрацаванай сістэмы ідэнтыфікацыі дыктара на ста дыктарах корпуса TIMIT пры выкарыстанні толькі мел-частотных кепстральных каэфіцыентаў у якасці спектральных прыкмет была атрыманая дакладнасць 96%, а пры выкарыстанні дадатковых каэфіцыентаў IMFCC яна дасягала 98-100%. Эксперыменты паказалі, што для гэтага датасэта выкарыстанне IMFCC у дадатак да MFCC значна (да 20%) павялічвае дакладнасць ідэнтыфікацыі, асабліва пры павелічэнні колькасці дыктараў у датасэце (гл. малюнак 5.2). Разам з тым для датасета LibriSpeech даданне IMFCC прывяло да паляпшэння якасці распазнання ўсяго на 1%.



Малюнак 5.2 – Залежнасць дакладнасці ідэнтыфікацыі ад колькасці дыктараў пры выкарыстанні MFCC і IMFCC (корпус TIMIT)

Таксама былі даследаваны розныя памеры вокнаў і частата іх перакрыцця пры першапачатковым разбіцці запісаў. На малюнку 5.3 подпіс выгляду 20/7 азначае, што бралася акно памерам 20 мс праз кожныя 7 мс. Высветлілася, што важна, каб вокны перакрываліся больш, чым напалову, а аптымальныя памеры акна і інтэрвалу яго ўзяцця складаюць 25 і 10 мс адпаведна.

Малюнак 5.3 – Залежнасць дакладнасці ідэнтыфікацыі ад памеру і кроку акна (корпус LibriSpeech)

Акрамя гэтага, быў даследаваны ўплыў формы акна на дакладнасць распазнавання (гл. табліцу 5.1). Стандартам лічыцца акно Хэмінга, таму ў якасці аналагаў былі абраны таксама вокны высокага разрознення падобнага вонкавага выгляду. У прынцыпе, усе яны паказалі супастаўныя вынікі, а выкарыстанне акна Ланцоша нават трохі павялічыла дакладнасць. Але так як яго вылічэнне даўжэй, пры разбіцці запісаў было вырашана выкарыстоўваць акно Хэмінга.

Табліца 5.1 – Залежнасць колькасці правільна распазнаных запісаў корпуса LibriSpeech ад формы акна

Назва акна	**Выраз у дыскрэтным выглядзе:** **w(n), n=0…N-1**	**Колькасць правільна распазнаных запісаў**
Акно Хэмінга (Hamming window)	$\mathrm{w(n)} = 0.54 - 0.46\cos\left(\frac{2\pi n}{N-1}\right)$	152/160 (95%)
Акно Ханна (Hann window)	$\mathrm{w(n)} = 0.5 - 0.5\cos\left(\frac{2\pi n}{N-1}\right)$	151/160 (94%)
Акно Блэкмана (Blackman window)	$\mathrm{w(n)} = a_0 - a_1\cos\left(\frac{2\pi n}{N-1}\right) + a_2\cos\left(\frac{4\pi n}{N-1}\right)$ $a_0 = 0.42, \quad a_1 = 0.5, \quad a_2 = 0.08$	152/160 (95%)
Акно Ланцоша (Lanczos window), ці sinc – окно	$\mathrm{w(n)} = \mathrm{sinc}\left(\frac{2n}{N-1} - 1\right), \ sinc(x) = \frac{\sin(\pi x)}{\pi x}$	153/160 (96%)
Сінус-акно	$\mathrm{w(n)} = \sin\left(\frac{\pi n}{N-1}\right)$	151/160 (94%)

Таксама быў даследаваны ўплыў на дакладнасць ідэнтыфікацыі дыктара розных статыстычных ацэнак сігналу, апісаных у [28], у якасці дадатковых прыкмет (гл. табліцу 5.2). Першапачаткова ім былі нададзены невялікія вагі. Так як большасць з іх не прывялі да паляпшэння дакладнасці ідэнтыфікацыі, не даследавалася іх уплыў з вялікім вагай. Выкарыстанне цэнтроіда амплітуды сігналу трохі павялічыла дакладнасць, хоць не настолькі, каб апраўдаць трату рэсурсаў на яго вылічэнне.

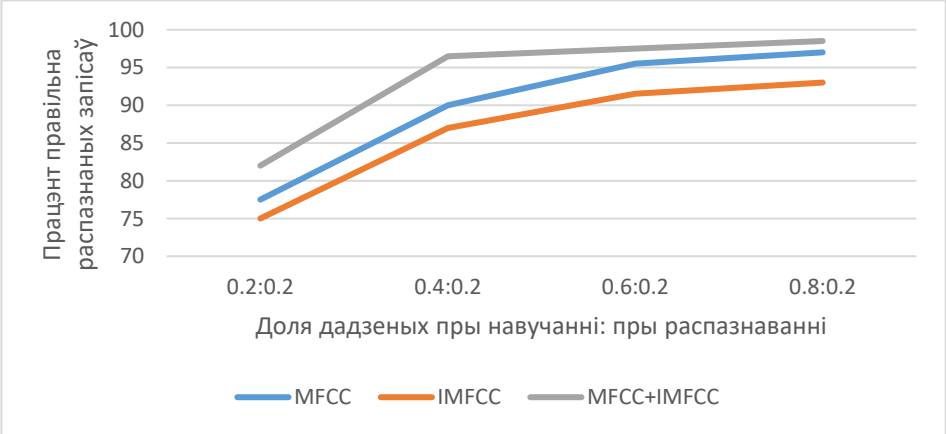
Табліца 5.2 – Залежнасць колькасці правільна распазнаных запісаў корпуса LibriSpeech ад вылічэння дадатковых статыстычных характарыстык сігналу

Дадатковая прыкмета	**Колькасць правільна распазнаных запісаў**
Толькі MFCC і іх вытворныя	1030/1062 (97%)
RollOff (спад амплітудна-частотнай характарыстыкі)	1027/1062 (97%)

Працяг табліцы 5.2.

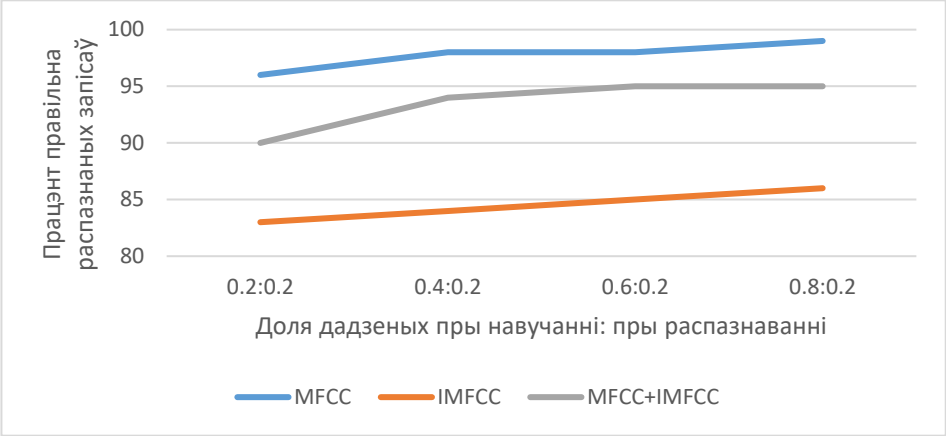
Дадатковая прыкмета	**Колькасць правільна распазнаных запісаў**
Zero Crossing Rate (частата перасячэння графіка амплітуды з нулём («частата пераходаў праз нуль» [17, с. 17]))	1028/1062 (97%)
Temporal Centroid (цэнтроід энергіі сігналу)	1031/1062 (97%)
Spectral Centroid (цэнтроід спектра сігналу)	1028/1062 (97%)
Root Mean Square (сярэднеквадратычнае значэнне спектру)	1028/1062 (97%)
Spectral Decrease (тэмп спаду спектральных значэнняў)	1027/1062 (97%)
Spectral Crest (стаўленне максімальнага значэння спектру да арыфметычнага сярэдняга)	1029/1062 (97%)
Spectral Flatness (стаўленне геаметрычнага сярэдняга да арыфметычнага)	1026/1062 (97%)

У адрозненне ад корпусаў англійскага маўлення, на корпусе беларускага маўлення даданне IMFCC прывяло да пагаршэння якасці распазнавання. Так, чыстыя MFCC далі дакладнасць 99%, а камбінацыя з IMFCC – толькі 95%. Для таго, каб праверыць, ці не залежаць вынікі дадання IMFCC ад аб'ёму дадзеных, даступных пры навучанні, былі праведзены эксперыменты са змяненнем гэтага аб'ёму на TIMIT (для ста дыктараў), вынікі якіх прадстаўлены на малюнку 5.4.



Малюнак 5.4 – Залежнасць дакладнасці ідэнтыфікацыі для розных камбінацый каэфіцыентаў ад колькасці дадзеных пры навучанні (корпус TIMIT)

Яны паказалі адсутнасць уплыву аб'ёму дадзеных, даступных пры навучанні, на перавагу адных каэфіцыентаў перад іншымі для абранага датасэта.



Малюнак 5.5 – Залежнасць дакладнасці ідэнтыфікацыі для розных камбінацый каэфіцыентаў ад колькасці дадзеных пры навучанні (корпус BelarusianSpeech)

Гэта было падцверджана і аналагічнымі эксперыментамі, якія былі праведзены з корпусам беларускага маўлення, як адлюстравана на малюнку 5.5.

Прычынай, па якой IMFCC могуць пагоршыць якасць распазнавання для распрацаванай мадэлі, можа быць іх карэляцыя з MFCC, а GMM мадэлі маюць на ўвазе некарэляванасць кампанентаў вектараў, на якіх яны будуюцца. У той жа час магчыма, што для корпуса TIMIT гэтая карэляцыя не такая моцная і дысперсія шуму ў цэлым менш. Таксама магчыма, што ў ангельскай мове больш, чым у беларускай, акцэнтаў, якія адрозніваюцца верхняй часткай спектру. Пацверджанне такой здагадкі апраўдала б выкарыстанне або невыкарыстанне IMFCC у залежнасці ад мовы датасета, але для гэтага патрабуюцца эксперыменты на вялікай колькасці датасэтаў розных моў.

6. Заключэнне

У ходзе праведзенай работы былі прааналізаваны існуючыя метады здабывання індывідуальных характарыстык чалавечага маўлення, спосабы пабудовы мадэляў дыктара і прыняцця рашэнняў для яго ідэнтыфікацыі. Была вывучана задача ідэнтыфікацыі дыктара, праблемы і абмежаванні, якія ўзнікаюць пры яе рашэнні, а таксама даследаваны сучасны сусветны рынак маўленчых тэхналогій і, у прыватнасці, існуючыя сістэмы ідэнтыфікацыі дыктараў, іх перавагі і недахопы, залежнасць ад мовы.

Таксама былі распрацаваны алгарытм тэкстанезалежнай ідэнтыфікацыі дыктара на аснове фанемнай сегментацыі і мадэляў гаусавых сумесяў і праграма, якая рэалізуе гэты алгарытм. Была створана акустычная база дадзеных для беларускай мовы, апрацаваны выбаркі запісаў мноства дыктараў з розных карпусоў маўлення і праведзены эксперыменты па ідэнтыфікацыі дыктараў пры розных параметрах сістэмы. Дакладнасць ідэнтыфікацыі склала 98% на ангельскім датасэце для ста дыктараў і 99% на беларускім датасэце для пяцідзесяці васьмі дыктараў пры найлепшых параметрах (акно Хэмінга або Ланцоша працягласцю 25 мс праз 10 мс, выкарыстанне цэнтроіда амплітуды сігналу, мадэляванне фанетычных GMM трыма гаусіанамі).

Перавагамі распрацаванай сістэмы з'яўляюцца:

- досыць высокая дакладнасць ідэнтыфікацыі дыктара;
- тэкстанезалежнасць;
- кросплатформеннасць сістэмы распазнавання дыктара;
- падтрымка мноства фарматаў гукавых файлаў;
- наяўнасць графічнага інтэрфейсу;
- лёгкасць дабаўлення ў сістэму новага дыктара.

Больш высокая якасць распазнавання ў параўнанні з традыцыйнымі тэкстанезалежнымі сістэмамі дасягаецца за кошт выкарыстання фанетычнай інфармацыі пры навучанні. Для існуючых сістэм, якія выкарыстоўваюць пабудову мадэляў гаусіан для фанем, пры ідэнтыфікацыі запіс таксама разбіваецца на фанемы, але гэта пагаршае якасць распазнавання ў выпадку, калі вымаўленага слова не было ў лексіконе, складзеным на аснове тэксту, вымаўленага дыктарамі пры навучанні. Разам з тым набор фанем, якія вымаўляюцца дыктарам, нашмат больш сталы, чым набор слоў. Працягласць нераспазнанай фанемы ў некалькі разоў менш, чым працягласць нераспазнанага слова, і адсутнасць яе ў слоўніку не так моцна адаб'ецца на якасці распазнання аўтара запісу, асабліва кароткай.

Устаноўлена, што ўплыў выкарыстання IMFCC ў якасці нізкаўзроўневых прыкмет на якасць распазнавання залежыць ад асаблівасцяў датасэта, у

прыватнасці, ад мовы, на якой гавораць дыктары. Таму пры распрацоўцы шматмоўных сістэм распазнання дыктараў пажадана ўлічваць асаблівасці мовы для павышэння якасці іх працы.

У будучыні плануецца працягнуць даследаванне ўплыву высокаўзроўневых характарыстык маўленчага сігналу на дакладнасць ідэнтыфікацыі, каб знайсці такія, якія дапамаглі б вылучыць аўтара запісу пры вялікай колькасці патэнцыйна магчымых дыктараў. У прыватнасці, плануецца даследаваць уключэнне фармантных частот у мадэлі фанем і магчымасць вылічэння частаты асноўнага тону дыктара на іх аснове, а таксама паэксперыментаваць з вылічэннем высокаўзроўневых характарыстык сігналу на прамежках канстантнай працягласці, якая перавышае памеры акна, але меншых, чым сярэдняя працягласць запісу. І, нарэшце, плануецца даследаваць эфектыўнасць ідэнтыфікацыі распрацаванай сістэмы на адкрытым мностве дыктараў і неабходнасць звязаных з гэтым мадыфікацый алгарытму прыняцця рашэнняў.

Спіс выкарыстаных крыніц

1. Сорокин В.Н., Вьюгин В.В., Тананыкин А.А. Распознавание личности по голосу. Аналитический обзор // Информационные процессы. – 2012. – Том 12, №1. – С. 1–30.
2. Рамишвили Г. С. Автоматическое опознавание говорящего по голосу. – М.: Радио и связь, 1981. – 224 с.
3. Лютова Д. А. Основные задачи и методы технологий распознавания говорящего по голосу // Вестник Московского государственного лингвистического университета: научно-технический журнал. - 2010. – Вып. 13. - С. 131–147.
4. Первушин Е.А. Обзор основных методов распознавания дикторов // Математические структуры и моделирование. – 2011. – Вып. 24. – С. 41–54.
5. Fusing High- and Low-Level Features for Speaker Recognition [Electronic resource]. – http://perso.telecom-paristech.fr/~chollet/Biblio/Cours/Biomet/Eurospeech03_SuperSIDFusionVfinal.pdf.
6. Швецов А.Г. Анатомия, физиология и патология органов слуха, зрения и речи: Учебное пособие. – Великий Новгород, 2006. – 68 с.
7. Signal Processing and Dynamic Time Warping. – http://www.ee.columbia.edu/~stanchen/fall12/e6870/slides/lecture2.pdf.
8. Christian Müller (Ed.). Speaker Classification I. – Springer-Verlag, Berlin Heidelberg, Germany, 2007. – P. 229–231.
9. S. Chakroborty and G. Saha. Improved Text-Independent Speaker Identification Using Fused MFCC and IMFCC feature Sets Based on Gaussian Filter // International Journal of Signal Processing. – 2009. – Vol. 5. No. 1. – P. 11–19.
10. Hynek Hermansky. Perceptual linear predictive (PLP) analysis of speech // J. Acoust. Soc. Am. –1990. – Vol. 87, No. 4. – P. 1738–1752.
11. Метод k-средних [Электрон. ресурс]. – https://ru.wikipedia.org/wiki/K-means.
12. EM-алгоритм [Электрон. ресурс]. – https://ru.wikipedia.org/wiki/EM-алгоритм.
13. Viterbi algorithm [Electronic resource]. – https://en.wikipedia.org/wiki/Viterbi_algorithm.
14. Comparative Study of Speaker Recognition Methods. DTW, GMM and SVM [Electronic resource]. – https://pdfs.semanticscholar.org/a143/9cc13f35aa5bf30b2a43071aedf84184aea9.pdf.
15. Gaussian Mixture Models and Introduction to HMM's [Electronic resource] – http://www.ee.columbia.edu/~stanchen/fall12/e6870/slides/lecture3.pdf.
16. Deep Neural Network Approaches to Speaker and Language Recognition [Electronic resource]. – https://groups.csail.mit.edu/sls/publications/2015/Dehak_IEEE-2015.pdf.

17. Voice Compare. Идентификация диктора по голосу [Электрон. ресурс]. – http://www.callcentre.by/index.php?area=1&p=static&page=voicecompare&print=1.
18. Corpus.by [Электрон. ресурс]. – http://corpus.by.
19. MLLR Transforms as Features in Speaker Recognition [Electronic resource]. – https://www.sri.com/sites/default/files/publications/mllr_transforms_as_features_in_speaker_recognition.pdf.
20. TranscriptionGenerator [Электрон. ресурс]. – https://ssrlab.grid.by/TranscriptionGenerator.
21. Combination of Features for Multilingual Speaker Identification with the Constraint of Limited Data [Electronic resource]. – https://pdfs.semanticscholar.org/c6d9/6088b98a41940b39285a143eb9d20d6b42ee.pdf.
22. Multilingual speaker recognition on Indian languages [Electronic resource]. – https://www.researchgate.net/publication/265314833_Multilingual_speaker_recognition_on_Indian_languages.
23. Text independent speaker identification in multilingual environments [Electronic resource]. – http://www.lrec-conf.org/proceedings/lrec2008/pdf/461_paper.pdf.
24. Speaker, accent, and language identification using multilingual phone strings [Electronic resource]. – https://www.ri.cmu.edu/pub_files/pub3/schultz_tanja_2002_1/schultz_tanja_2002_1.pdf.
25. Nautsch, A. Speaker verification using i-vectors: M.Sc. thesis / Hochschule Darmstadt, University of Applied Science, 2014. – 131p.
26. Dehak, N. [et al.] Front-End Factor Analysis for Speaker Verification // IEEE transactions on audio, speech, and language processing. – 2011. – Vol. 19. No. 4. – P. 788–797.
27. Comparing Open-Source Speech Recognition Toolkits [Electronic resource]. – http://suendermann.com/su/pdf/oasis2014.pdf.
28. Kaldi [Electronic resource]. – http://kaldi-asr.org.

ДАДАТАК А

ПРАГРАМНЫЯ ЛІСТЫНГІ

Лістынг А.1 – Ініцыялізацыя мадэляў дыктараў

```
void SpeakerModeler::BuildDictorModels(const map<int, map<int, vector<vector<double>>>>& features) const {
    map<int, map<int, gmm_diag>> models;
    for (auto& kv : features) {
        for (auto& fkv : kv.second) {
            if (fkv.second.size() >= gaussians) {
                gmm_diag model;
                mat F(fkv.second);
                if (model.learn(F, gaussians, maha_dist, random_subset, 10, 100, 1e-4, false)) {
                    models[kv.first][fkv.first] = model;
                }
            }
        }
    }
    WriteModel(modelFolder, modelPath, models);
}
```

Лістынг А.2 – Вылічэнне прыкмет запісу маўлення

```
void SpeakerModeler::ExtractBatchFeatures(const string& folder, const string& alignment_path,
    map<int, map<int, vector<vector<double>>>>& features, map<int, string>& files) {
    string line;
    int phoneme, record, dictor, prev_record = -1;
    int window_size_in_samples, hop_in_samples, sampling_rate, offset;
    const double window_offset = 0.25;
    double start, end;
    SoundFile sound;
    vector<double>& data = sound.getData();
    std::ifstream f(alignment_path);
    if (f.is_open()) {
        while (getline(f, line)) {
            SSCANF(&line[0], "%d %d %lf %lf %d", &dictor, &record, &start, &end, &phoneme);
            if (prev_record != record) {
                if (!sound.Initialize(folder + files[record])) {
                    fprintf(stderr, "Cannot parse file %s\n", (folder + files[record]).c_str());
                    return;
                }
                sampling_rate = sound.getSamplingRate();
                window_size_in_samples = windowSize * sampling_rate;
                offset = window_size_in_samples * window_offset;
                hop_in_samples = hop * sampling_rate;
                data = sound.getData();
                prev_record = record;
                printf("%d\n", record);
            }
            int start_in_samples = start * sampling_rate - offset;
            start_in_samples = std::max(start_in_samples, 0);
            int end_in_samples = end * sampling_rate + offset;
            end_in_samples = std::min(end_in_samples, static_cast<int>(data.size())) - window_size_in_samples;
            for (int i = start_in_samples; i < end_in_samples; i += hop_in_samples) {
                vector<double> input;
                input.reserve(window_size_in_samples);
```

```
                for (int j = 0; j < window_size_in_samples; ++j)
                {
                    input.push_back(data[i + j] * windowFunc(j, window_size_in_samples));
                }
                vector<double> segment_result;
                for (const auto &extractor : extractors)
                {
                    extractor->Extract(segment_result, input, sound);
                }
                features[dictor][phoneme].push_back(segment_result);
            }
        }
        f.close();
    }
    else {
        printf("Cannot open file %s\n", alignment_path.c_str());
    }
}
```

Лістынг А.3 – Вызначэнне дыктара па вылучаных прыкметах

```
void SpeakerRecognizer::ExtractFeatures(const string& path, vector<vector<double>>& tests) {
	SoundFile sound;
	if (!sound.Initialize(path)) {
		fprintf(stderr, "Cannot parse file %s\n", path.c_str());
		return;
	}
	int sampling_rate = sound.getSamplingRate();
	double window_size_in_samples = windowSize * sampling_rate;
	double hop_in_samples = hop * sampling_rate;
	const vector<double>& data = sound.getData();
	int end = data.size() - window_size_in_samples;

	for (int i = 0; i < end; i += hop_in_samples) {
		vector<double> input;
		for (int j = 0; j < window_size_in_samples; ++j) {
			input.push_back(data[i + j] * windowFunc(j, window_size_in_samples));
		}
		vector<double> segment_result;
		mfccExtractor.Extract(segment_result, input);
		tests.push_back(segment_result);
	}
}

int best_speaker(map<int, int>&result) {
	int max = 0;
	int best = 0;
	for (auto& kv : result) {
		if (kv.second > max) {
			max = kv.second;
			best = kv.first;
		}
	}
	return best;
}

int SpeakerRecognizer::TellSpeaker(const vector<vector<double>>& features) const {
	int best;
	map<int, int> result;
	for (auto& kv : dictors) {
		result[kv.first] = 0;
	}
	for (int i = 0; i < features.size(); i++) {
		double cur, max = -DBL_MAX;
		for (auto& kv : dictors) {
			for (auto& model : kv.second) {
				mat F(features[i]);
				cur = model.avg_log_p(F);
				if (cur > max) {
					max = cur;
					best = kv.first;
				}
			}
		}
		result[best]++;
	}
	return best_speaker(result);
}
```

Лістынг А.4 – Вылічэнне мел-частотных кепстральных каэфіцыентаў

```
void MfccExtractor::Extract(vector<double>& feature_vec, vector<double>& input) {
        int size = input.size();

        fftw_complex* in(static_cast<fftw_complex*>(fftw_malloc(sizeof(fftw_complex) * size)));
        fftw_complex* out(static_cast<fftw_complex*>(fftw_malloc(sizeof(fftw_complex) * size)));
        fftw_plan p(fftw_plan_dft_1d(input.size(), in, out, FFTW_FORWARD, FFTW_ESTIMATE));
        for (size_t j = 0; j < size; ++j) {
                in[j][0] = input[j];
                in[j][1] = 0;
        }
        fftw_execute(p);
        size /= 2;
        double* power_spectrum = new double[size];
        for (int i = 0; i < size; ++i) {
                power_spectrum[i] = sqrt(powf(out[i][0], 2) + powf(out[i][1], 2));
        }
        fftw_destroy_plan(p);
        fftw_free(in);
        fftw_free(out);

        for (int i = 0; i < mfccExtractors.size(); i++) {
                mfccExtractors[i]->GetLogCoefficents(power_spectrum, feature_vec);
        }
        delete[] power_spectrum;
}
```

Лістынг А.5 – Прывядзенне файлаў Калдзі да ўнутранага фармату сістэмы пры разбіцці на фанемы запісаў дыктара

```
#!/usr/bin/env perl

if (@ARGV != 3) {
  print STDERR "Usage: format_ali_single.pl <ctm-alignment-file> <record_number> <result-file>\n";
  print STDERR "Example: format_ali_single.pl exp/tri4a_ali/ali.1.ctm 1 result_ali.1.txt\n";
  exit(1);
}

$ctm_alignment = shift @ARGV;
$rec_number = shift @ARGV;
$result_alignment = shift @ARGV;

open $input, $ctm_alignment or die "Could not open $ctm_alignment: $!";
open $output, ">$result_alignment" or die "Could not open '$result_alignment' $!";
while($line = <$input>) {
  @words = split / /, $line;
  $end = @words[2] + @words[3];
  print $output "0 ".$rec_number." ".@words[2]." ".$end." ".@words[4];
}
close $input;
close $output;
```

Лістынг А.6 – Пабудова лінгвістычнай мадэлі і разбіццё на фанемы

```
#!/bin/bash
. ./cmd.sh

NJ=8

mfccdir=mfcc/train
x=data/train
./steps/make_mfcc.sh --cmd "$train_cmd" --nj $NJ $x exp/make_mfcc/$x $mfccdir
./steps/compute_cmvn_stats.sh $x exp/make_mfcc/$x $mfccdir

./steps/train_mono.sh --nj $NJ --cmd "$train_cmd" data/train data/lang exp/mono;
./steps/align_si.sh --nj $NJ --cmd "$train_cmd" data/train data/lang exp/mono
exp/mono_ali;

./steps/train_deltas.sh --cmd "$train_cmd" 2000 10000 data/train data/lang exp/mono_ali
exp/tri1;
./steps/align_si.sh --nj $NJ --cmd "$train_cmd" data/train data/lang exp/tri1
exp/tri1_ali;
./steps/train_deltas.sh --cmd "$train_cmd" 2500 15000 data/train data/lang exp/tri1_ali
exp/tri2a;
./steps/align_si.sh --nj $NJ --cmd "$train_cmd" --use-graphs true data/train data/lang
exp/tri2a exp/tri2a_ali;
./steps/train_lda_mllt.sh --cmd "$train_cmd" 3500 20000 data/train data/lang
exp/tri2a_ali exp/tri3a;
./steps/align_fmllr.sh --nj $NJ --cmd "$train_cmd" data/train data/lang exp/tri3a
exp/tri3a_ali;
./steps/train_sat.sh --cmd "$train_cmd" 4200 40000 data/train data/lang exp/tri3a_ali
exp/tri4a;
```

Лістынг А.7 – Пабудова мадэляў дыктараў

```
nj=8
cmd=utils/run.pl
program=./SpeakerModeler

. utils/parse_options.sh

if [ $# -lt 5 ] || [ $# -gt 6 ]; then
   echo "Usage: $0 <model-folder> <alignment-path> <record-folder> <record-location-file> [<log-dir>]";
   echo "Note: <log-dir> defaults to <model-folder>/log";
   echo "e.g.: $0 models data/train/result_ali data/LibriSpeech/dev-clean data/train/rec_map.txt";
   echo "Options: " ;
   echo "  --nj <nj>                                   # number of parallel jobs";
   echo "  --cmd (utils/run.pl|utils/queue.pl <queue opts>) # how to run jobs.";
   exit 1;
fi

if [ $# -ge 6 ]; then
  logdir=$5
else
  logdir=$model_folder/log
fi

model_folder=$1
ali_path=$2
record_folder=$3;
record_map=$4;

rm $logdir/*;
rm $model_folder/*;

$cmd JOB=1:$nj $logdir/speaker_modeler.JOB.log $program $model_folder/ model.JOB.txt ${ali_path}.JOB $record_folder/ $record_map;
echo "Dictor models have been successfully written to $model_folder.";
```

Лістынг А.8 – Распазнаванне дыктараў

```
program=./SpeakerRecognizer;

if [ $# -lt 5 ] || [ $# -gt 6 ]; then
   echo "Usage: $0 <model-folder> <record-folder> <test-file> <result-file> [<log-dir>]";
   echo "Note: <log-dir> defaults to <model-folder>/log";
   echo "e.g.: $0 models data/LibriSpeech/test-clean test_files.txt recognized.txt";
   exit 1;
fi
if [ $# -ge 6 ]; then
  logdir=$5
else
  logdir=$model_folder/log
fi
rm $logdir/*;

model_folder=$1
record_folder=$2;
test_files=$3;
result=$4;

tmpModelFile="$model_folder/tmp_model.txt";
modelFile="$model_folder/model.txt";

if [ -f $tmpModelFile ] ; then
    rm $tmpModelFile
fi
if [ -f $modelFile ] ; then
    rm $modelFile
fi

cat $model_folder/model.*.txt > $tmpModelFile;
./cat_models.pl $tmpModelFile $modelFile;

$program $result $model_folder model.txt $test_files $model_folder;
echo "Result has been successfully written to $result";
```

Лістынг А.9 – Разбіццё на фанемы запісаў новага дыктара на аснове існуючай лінгвістычнай мадэлі

```
#!/bin/bash

if [ $# -lt 4 ]; then
  echo "Usage: process.sh <kaldi-trunk> <source-folder> <user-id> <record-id>"
  exit 1
fi

kaldi=$1
source=$2
id=$3
rec_number=$4
steps=${kaldi}/egs/wsj/s5/steps
utils=${kaldi}/egs/wsj/s5/utils
src=${kaldi}/src

export train_cmd="run.pl --mem 2G"
export decode_cmd="run.pl --mem 4G"
export mkgraph_cmd="run.pl --mem 8G"

export KALDI_ROOT=${kaldi}
[ -f $KALDI_ROOT/tools/env.sh ] && . $KALDI_ROOT/tools/env.sh
export PATH=${utils}:$PATH
[ ! -f $KALDI_ROOT/tools/config/common_path.sh ] && echo >&2 "The standard file
$KALDI_ROOT/tools/config/common_path.sh is not present -> Exit!" && exit 1
. $KALDI_ROOT/tools/config/common_path.sh
export LC_ALL=C

pushd ${source}

ln -sf ${utils}/ .

if [[ ${source} == *en_Timit ]]; then
  base=${source/en_Timit/en_LibriSpeech}
  ln -sf ${base}/exp/tri4a exp/tri4a
  ln -sf ${base}/data/lang data/lang
fi

x=data/${id}
ali=exp/${id}

${steps}/make_mfcc.sh --cmd "$train_cmd" --nj 1 $x exp/make_mfcc/$x mfcc/${id} && \
${steps}/compute_cmvn_stats.sh $x exp/make_mfcc/$x mfcc/${id} && \
${steps}/align_fmllr.sh --nj 1 --cmd "$train_cmd" $x data/lang exp/tri4a ${ali} && \
${src}/bin/ali-to-phones --ctm-output exp/tri4a/final.mdl ark:"gunzip -c
${ali}/ali.1.gz|" -> ${ali}/ali.1.ctm && \
./format_ali_single.pl ${ali}/ali.1.ctm $x/ali.${rec_number} || { popd; exit 1; }

popd
exit 0;
```

ДАДАТАК В

СТРУКТУРА ФАЙЛАЎ ДЛЯ ПАБУДОВЫ ЛІНГВІСТЫЧНАЙ МАДЭЛІ

```
1089-134686-0000 HE HOPED THERE WOULD BE STEW FOR DINNER TURNIPS AND CARROTS AND BRUISED
1089-134686-0001 STUFF IT INTO YOU HIS BELLY COUNSELLED HIM
1089-134686-0002 AFTER EARLY NIGHTFALL THE YELLOW LAMPS WOULD LIGHT UP HERE AND THERE THE
1089-134686-0003 HELLO BERTIE ANY GOOD IN YOUR MIND
```

Малюнак В.1 – Файл text – тэкставая транскрыпця запісаў

```
1089-134686 m
1089-134691 m
1188-133604 m
121-121726 f
121-123852 f
121-123859 f
121-127105 f
```

Малюнак В.2 – Файл spk2gender з пазнакай пола дыктараў

```
1089-134686-0000 1089-134686
1089-134686-0001 1089-134686
1089-134686-0002 1089-134686
1089-134686-0003 1089-134686
1089-134686-0004 1089-134686
1089-134686-0005 1089-134686
1089-134686-0006 1089-134686
```

Малюнак В.3 – Файл utt2spk с пазнакай дыктара для кожнага запісу

```
134686-0000 flac -c -d -s /home/victoria/LibriSpeech/dev-clean/1089/134686/1089-134686-0000.flac |
134686-0001 flac -c -d -s /home/victoria/LibriSpeech/dev-clean/1089/134686/1089-134686-0001.flac |
134686-0002 flac -c -d -s /home/victoria/LibriSpeech/dev-clean/1089/134686/1089-134686-0002.flac |
134686-0003 flac -c -d -s /home/victoria/LibriSpeech/dev-clean/1089/134686/1089-134686-0003.flac |
134686-0004 flac -c -d -s /home/victoria/LibriSpeech/dev-clean/1089/134686/1089-134686-0004.flac |
134686-0005 flac -c -d -s /home/victoria/LibriSpeech/dev-clean/1089/134686/1089-134686-0005.flac |
```

Малюнак В.4 – Файл wav.scp з пазнакай шляхоў да запісаў з магчымасцю канвертавання "на ляту", выкарыстоўваючы камандныя канвееры

```
<oov>
AA0
AA1
AA2
AE0
AE1
AE2
```

Малюнак В.5 – Файл nonsilence_phones.txt з фанемамі, якія не пазначаюць цішыню

```
SIL
oov
```

Малюнак В.6 – Файл silence_phones.txt з фанемамі, якія пазначаюць цішыню

```
SIL
```

Малюнак В.7 – Файл optional_silence.txt

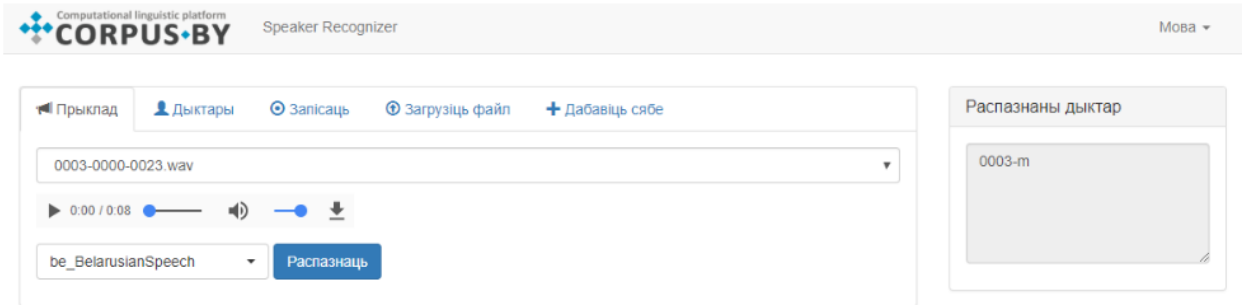
```
<oov> <oov>
DOG D AO1 G
DOING D UW1 IH0 NG
DOLL D AA1 L
```

Малюнак В.7 – Файл lexicon.txt з транскрыпцыяй слоў лексікону

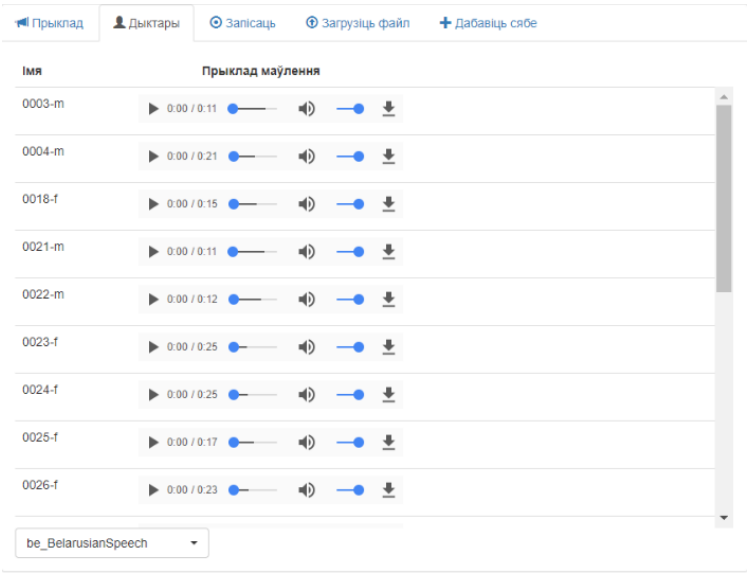
ДАДАТАК С

СКРЫНШОТЫ ВЭБ-СЭРВІСУ

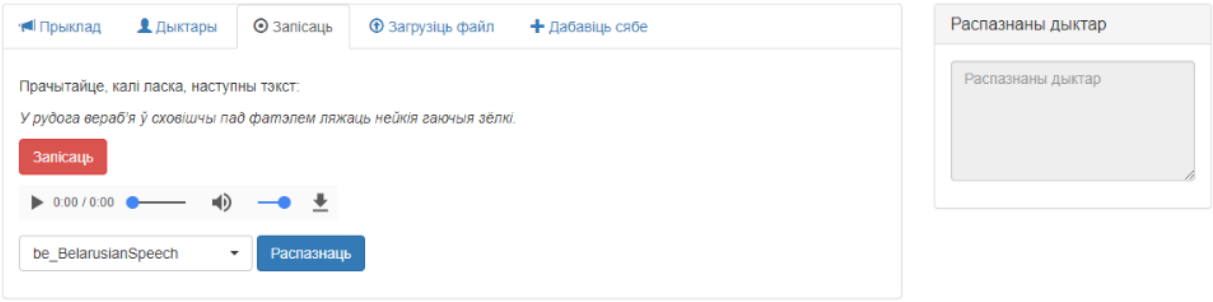
Малюнак С.1 – Укладка Example



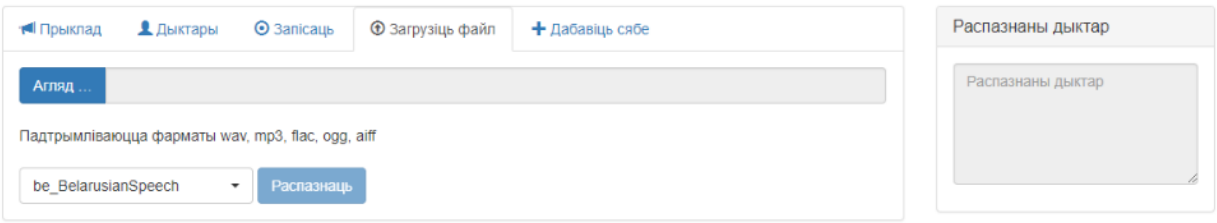
Малюнак С.2 – Укладка Speakers



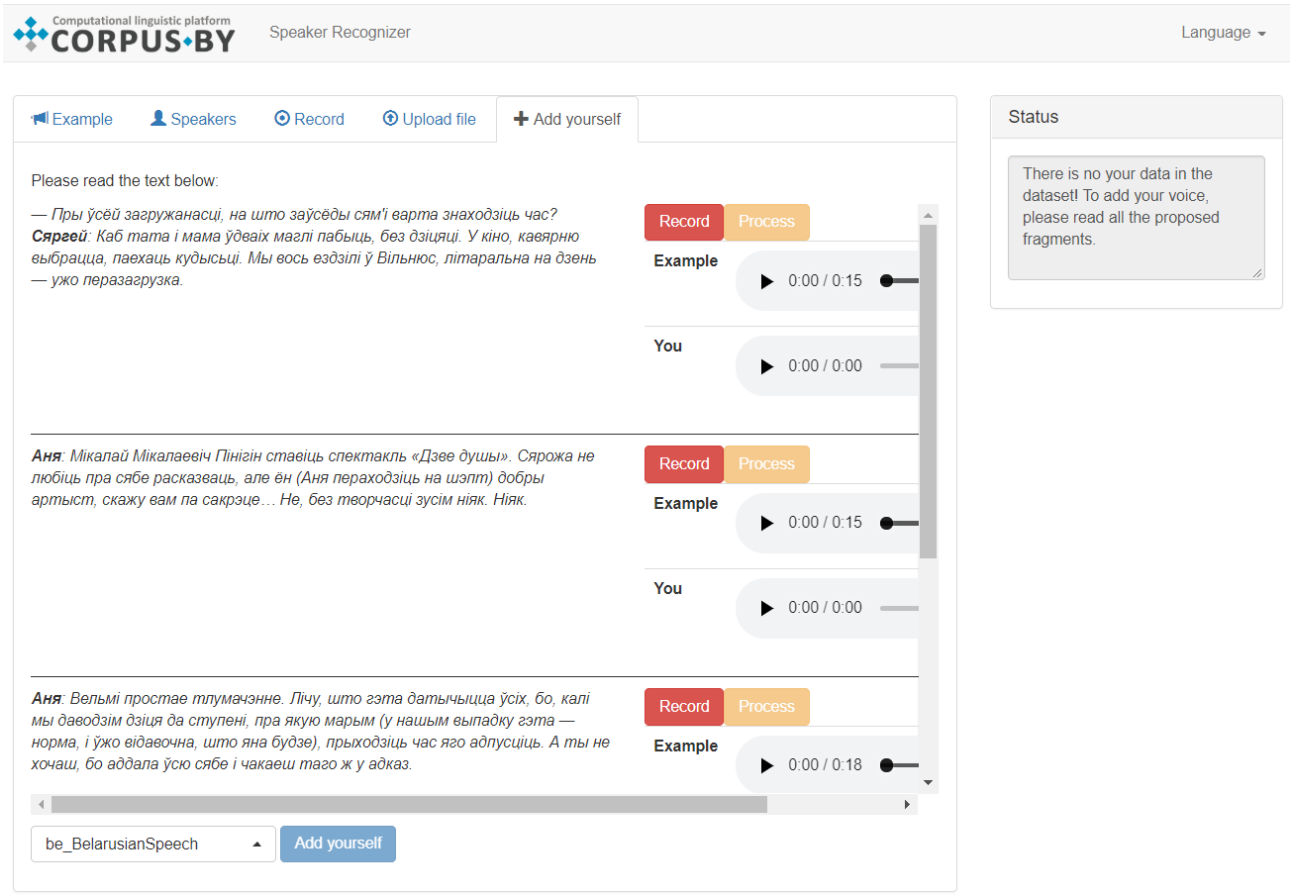
Малюнак С.2 – Укладка Record



Малюнак С.3 – Укладка Upload



Малюнак С.4 – Укладка Add yourself



Малюнак С.5 – Выбар датасэта