УДК 004.89 : 004.52

О.В. ПАВЛОВЕЦЬ, Є.С. СУЛЕМА
Національний технічний університет України «Київський політехнічний інститут»

МЕТОД РОЗПІЗНАВАННЯ МОВЛЕННЄВИХ ОДИНИЦЬ НА ОСНОВІ ADABOOST

В статті розглядаються особливості побудови системи розпізнавання мови на основі AdaBoost класифікатора даних. В результаті проведеного дослідження запропоновано метод розпізнавання мовленнєвих одиниць з використанням модифікації MFCC дескриптора. В статті представлені результати розпізнавання із застосуванням лінійного SVM, а також статистичні дані щодо якості розпізнавання.

Ключові слова: AdaBoost, SVM, система розпізнавання мови, MFCC, фонема.

OLEKSANDR V. PAVLOVETS, YEVGENIYA S. SULEMA
National Technical University of Ukraine "Kyiv Polytechnic Institute"

METHOD OF SPEECH UNITS' RECOGNITION BASED ON ADABOOST

Abstract – This paper concerns the task of the development of speech units' recognition system based on AdaBoost method that is one of the most popular machine learning algorithms. The analysis of speech units (groups of phonemes) and their spectrums is fulfilled and presented. The architecture of the system is presented and discussed. The system includes: 1. Module for speech enhancement; 2. Module for descriptor creation; 3. Module for recognition; 4. Module for heuristics. The module for speech enhancement is used for noise reduction. Both the module of descriptor creation and the module for recognition are based on using of modified Mel Frequency Cepstral Coefficients (MFCC) descriptor. The proposed descriptor consists of 51 values and corresponds to 50 ms of audio signal. Linear Support Vector Machine (SVM) classifier is used as well. The module for heuristics is used for recognition results' enhancement and based on texts' pre-analysis. Both the experimental results and statistical data of recognition quality depending on various system parameters are presented in the paper.

Keywords: AdaBoost, SVM, speech recognition system, MFCC, phoneme

Вступ

В наш час бурхливий розвиток інформаційних технологій ставить все нові вимоги до інтерфейсу користувача інтелектуальних систем. Одна з таких вимог – наблизити спілкування людини з машиною до природного, інтуїтивно зрозумілого й притаманного людні рівня. Таким інтерфейсом може стати мова – основний засіб спілкування між людьми. Дослідження в галузі автоматичного розпізнавання мови ведуться вже більш, ніж 50 років. Кінцевою метою таких досліджень є побудова комп'ютерної системи, яка не гірше за людину зможе розпізнавати слова та розуміти їх значення. Потреба в таких системах і зацікавленість великої кількості дослідників дає свої результати: на сьогодні відома чимала кількість успішних спроб реалізувати систему розпізнавання мови. Але, не дивлячись на помітні успіхи у цій галузі, дослідники поки що далекі від того, щоб комп'ютерна система правильно інтерпретувала сенс мови на довільну тему, яку виконує довільний диктор в довільному навколишньому оточенні.

Аналіз основних досягнень в області розпізнавання мови [1] дозволяє виділити чималу кількість задач, які на теперішній час не мають чіткого і однозначного вирішення, тому дослідження у сфері систем розпізнавання мови у наш час є актуальними, причому інтерес до таких продуктів, як і потреба в них, неухильно зростає.

Терміни та визначення

Фонема – найменша структурно-семантична звукова одиниця мови.

Мовленнєва одиниця – звичайно в системах розпізнавання мови під мовленнєвою одиницею розуміють деяку фонему; в даній роботі цей термін використовується для позначення певної групи схожих за звучанням фонем, що розпізнаються системою як єдиний клас.

Акустичний сигнал – деякий мовний сигнал, який несе певне змістове навантаження.

Дескриптор акустичного сигналу – певний вектор, який надає компактне та інформативне представлення акустичного сигналу.

MFCC (Mel Frequency Cepstral Coefficients) – Мел-кепстральні коефіцієнти [2].

Mel Scale – шкала залежності гучності звуку від частоти коливань, заснована на психофізичній одиниці висоти звуку *мел* [3].

Pre-emphasis – попередня частотна корекція, що використовується для згладжування перепадів амплітуди в сигналі.

Вікно – деяка частина сигналу фіксованої довжини.

DFT (Discrete Fourier Transform) – дискретне перетворення Фур`є.

DCT (Discrete Cosine Transform) – дискретне косинусне перетворення.

SVM (Support Vector Machine) – метод опорних векторів, один з найбільш популярних методів машинного навчання за прецедентами; в даному досліджені використовувався лінійний *SVM* (далі – *LSVM*).

AdaBoost (Adaptive Boosting) – один із найпопулярніших алгоритмів машинного навчання, який передбачає побудову композиції класифікаторів; в даному досліджені використовувався AdaBoost на основі дерева рішень.

Класифікація об'єкту – віднесення об'єкту до певного класу, який визначається номером або найменуванням.

Постановка задачі

В даній роботі ставиться задача побудувати систему розпізнавання значущих одиниць мови на основі AdaBoost класифікатора даних та LSVM із застосуванням евристичного алгоритму підвищення якості розпізнавання.

Архітектура системи

Розглянемо узагальнену систему розпізнавання мовних одиниць (рис.1). На першому етапі роботи системи виконується виділення змістових фрагментів з вхідного сигналу (наприклад, з неперервного потоку, що отримується з мікрофону). Для цього можуть бути застосовані різного роду детектори голосової активності (VAD – Voice Activity Detector). Отриманий акустичний сигнал фільтрується з метою заглушення шумів у блоці видалення шумів, який ще називають блоком покращення мови (Speech Enhancement [4]).

Слід зазначити, що всі експерименти, що виконувались у рамках даного дослідження, проводились над набором сигналів (40 звукових файлів з фонетичною транскрипцією [5]), що характеризуються низьким рівнем зашумленості, тому в проведених експериментах методи видалення шумів не використовувалися.

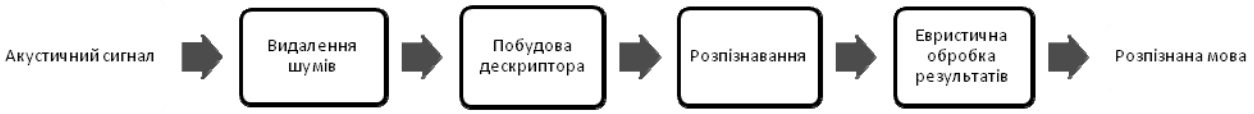

Рис. 1. Принцип роботи системи розпізнавання мовленнєвих одиниць

Метою роботи блока побудови дескриптора є отримання певного вектора, який є компактним і достатньо інформативним представленням акустичного сигналу. Отже, головне завдання на цьому етапі роботи системи – надати акустичному сигналу компактну форму, яка б містила чітко виражену лінгвістичну складову.

Блок розпізнавання є основною складовою системи розпізнавання мовленнєвих одиниць. Результатом його роботи є послідовність символів, що відповідає вхідному мовленнєвому сигналу. З метою підвищення якості розпізнавання сигналу до складу системи введено блок евристичної обробки результатів, у якому виконується аналіз гіпотез щодо змістової складової мовленнєвого сигналу. Розглянемо окремі складові системи розпізнавання мовленнєвих одиниць детальніше.

Побудова дескриптора

На сьогодні існує чимало ефективних дескрипторів. Всі вони будуються з врахуванням принципів роботи слухової системи людини і сприйняття людиною мовленнєвих сигналів. В даному досліджені були використані MFCC. Основні переваги їх застосування:

- врахування особливостей сприйняття акустичних сигналів людиною;
- стійкість до шумів;
- стійкість до незначних відхилень вікна;
- дескриптор на основі MFCC достатньо інформативний для розпізнавання окремих фонем.

На рис. 2 представлений узагальнений алгоритм розрахунку MFCC. Для розпізнавання мовленнєвих сигналів достатньо перших 13 коефіцієнтів [6].

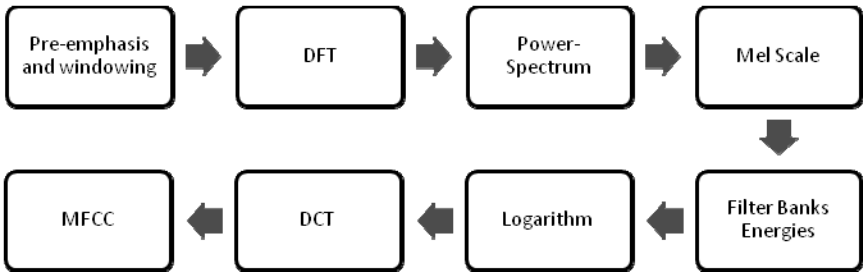

Рис. 2. Алгоритм розрахунку MFCC

Після аналізу спектрів фонем, виділених у базі даних [5], що використовувалась у даному досліджені, було вирішено розбити їх на 3 групи. До першої групи фонем віднесено «стабільні в часі фонеми». Як видно з рис. 3, спектр таких сигналів показує чіткі періодичності в часі (ліва колонка). MFCC розраховувалися з наступними параметрами: кількість банків фільтрів – 20, кількість кепстральних коефіцієнтів – 13, мінімальна частота – 25 Гц, максимальна частота – 6855.6 Гц, довжина вікна – 25 мс, довжина перекриття вікон – 12,5 мс.

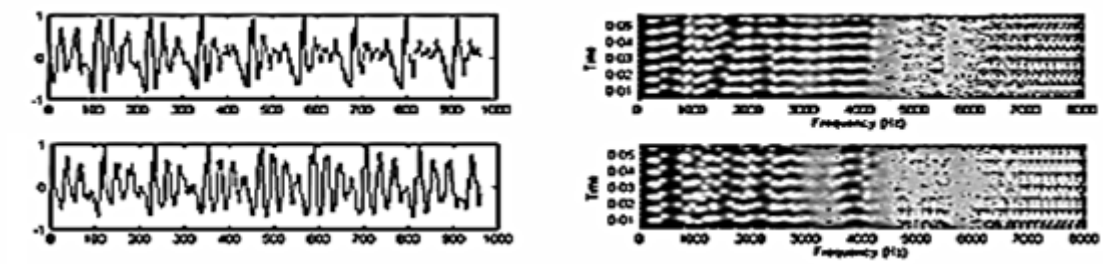

Рис. 3. Два екземпляри фонеми «а». Ліва колонка – сигнал, права колонка – спектр сигналу

До «стабільних в часі» фонем відносяться всі голосні фонеми. Також до цього класу фонем можна віднести і сонорні приголосні, включаючи «б», «д», «в» і «г», а також пом'якшені версії цих фонем. Як показано на рис. 4, пом'якшені версії «стабільних в часі» фонем дуже схожі з твердими аналогами. Основна відмінність полягає в тому, що в кінці звучання відбувається так зване «пом'якшення», що визначається у спектрі сплеском частот в діапазоні 3000–4000 Гц.

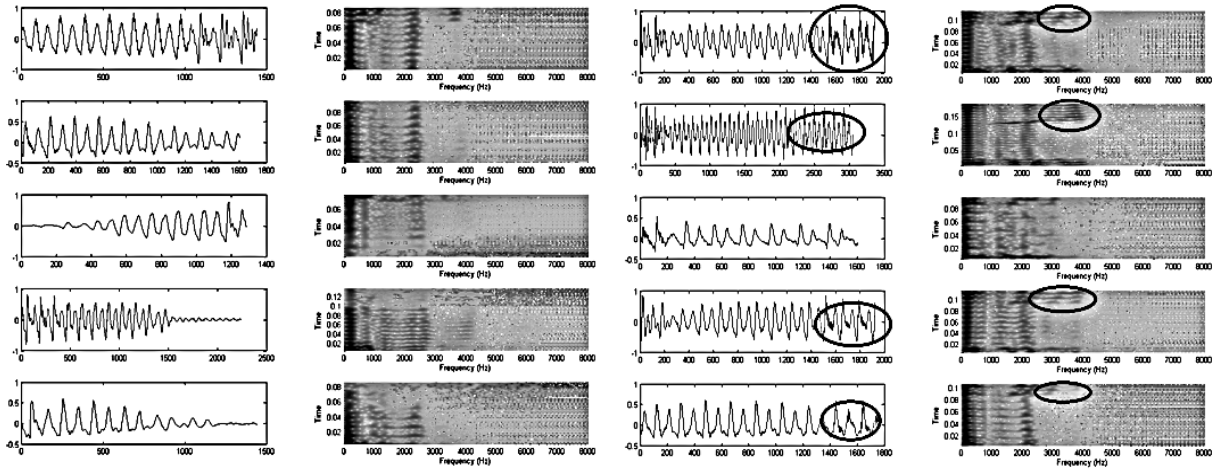

Рис. 4. Порівняння сигналів і спектрів фонем «м» і «м'»

До наступного класу фонем відносяться так звані «шумові» фонеми. На спектрі таких фонем (рис. 5) не відслідковується жодних яскраво виражених періодичностей, тому цей клас отримав назву «нестабільних в часі» фонем.

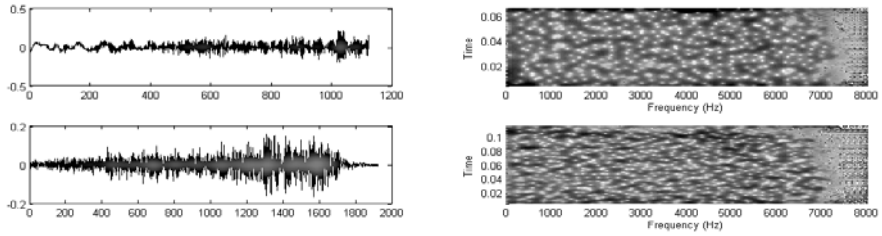

Рис. 5. Два екземпляри фонеми «с». Ліва колонка – сигнал, права колонка – спектр сигналу

До останнього, третього класу було віднесено такі особливі фонеми, як «к», «п», «т», та їх пом'якшені версії. Як видно з рис. 6, це також «нестабільні в часі» фонеми, але їх особливість полягає в тому, що лінгвістична інформаційна складова сигналу знаходиться в кінці, тобто сигнал можна поділити умовно на дві частини: паузу і короткий сплеск. Таку особливість цих сигналів безумовно потрібно враховувати при генерації даних для навчання системи.

Враховуючи особливості виділених трьох класів фонем була запропонована структура дескриптора, наведена на рис. 7. Найменшою довжиною сигналу, на якій можливо побудувати дескриптор є 50 мс. Як було сказано вище, MFCC розраховуються у вікні в 25 мс з перекриттям в 12,5 мс, отже, з сигналу в 50 мс можна згенерувати 3 дескриптори MFCC по 13 коефіцієнтів кожен, як показано на рис. 7. Спектр сигналу кожного вікна MFCC ділиться на 4 рівні проміжки, для кожного з яких розраховується енергія. Далі ці значення додаються до дескриптора. Таким чином, отримуємо для сигналу в 50 мс набір, наведений на рис. 8. Загальна довжина такого дескриптора становить 51 значення.

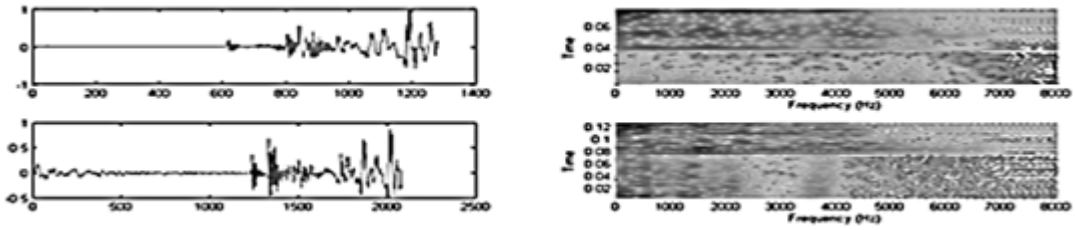

Рис. 6. Два екземпляри фонеми «к». Ліва колонка – сигнал, права колонка – спектр сигналу

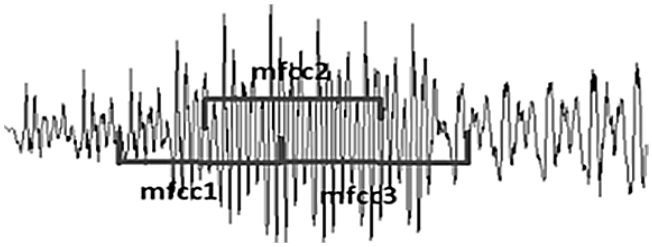

Рис. 7. Структура дескриптора на основі трьох вікон з MFCC

$E_{1,1}E_{1,2}E_{1,3}E_{1,4}$	$MFCC_1$	$E_{2,1}E_{2,2}E_{2,3}E_{2,4}$	$MFCC_2$	$E_{3,1}E_{3,2}E_{3,3}E_{3,4}$	$MFCC_3$

Рис. 8. Набір елементів дескриптора

Генерація даних для навчання системи полягає у побудові подібних дескрипторів для кожного екземпляру окремої фонеми. Довжина екземпляру фонеми не є сталим значенням, але для застосування певного класифікатору для кожної фонеми повинен бути згенерований дескриптор однакового сталого розміру. Для вирішення цієї проблеми була застосована наступна методика. Якщо довжина екземпляру фонеми (екземпляр фонеми отримуємо з бази даних що містить розмітку, тобто позначки початку і кінця звучання фонеми) менше 50 мс, то для навчання системи береться сигнал доповнений зліва і справа до 50 мс. В наявній базі даних таких екземплярів фонем дуже мало, як правило їх довжина (час звучання) значно перевищує 50 мс. Якщо довжина екземпляру фонеми перевищує 50 мс, то цей сигнал також доповнюється зліва і справа до довжини кратній 12,5 мс (довжина перекриття вікон). Це робиться для того, щоб для кожного із екземплярів фонем можна було згенерувати ціле число вікон для MFCC. Таким чином, якщо довжина деякого екземпляру фонеми вирівняна на границю 12,5 мс, отримуємо деяку кількість вікон з MFCC (рис. 9). Далі виконуємо декомпозицію за правилом, наведеним на рис. 10.

$MFCC_1$	$MFCC_2$	$MFCC_3$	$MFCC_4$	$MFCC_5$

Рис. 9. Приклад обчислення MFCC для екземпляру фонеми довжиною 75 мс

$E_{1,i}$	$MFCC_1$	$E_{2,i}$	$MFCC_2$	$E_{3,i}$	$MFCC_3$
$E_{2,i}$	$MFCC_2$	$E_{3,i}$	$MFCC_3$	$E_{4,i}$	$MFCC_4$
$E_{3,i}$	$MFCC_3$	$E_{4,i}$	$MFCC_4$	$E_{5,i}$	$MFCC_5$

Рис. 10. Правило створення екземплярів для навчання з фонеми довжиною 75 мс

Таким чином, з одного екземпляру фонеми отримуємо декілька дескрипторів сталої довжини, які трактуються як окремі екземпляри звучання однієї і тієї ж фонеми. Але, щоб таке подання дескрипторів було коректним, необхідно врахувати особливості кожного із трьох виділених класів фонем. Для перших двох класів («стабільні в часі» і «нестабільні в часі» шумові фонеми) генерація дескрипторів буде коректною (для першого класу MFCC протягом всього звучання сигналу з екземпляром фонеми мають схожі значення, для другого класу генерація дескрипторів подібного роду допоможе однаково розпізнавати початок, середину і кінець такої фонеми), але для третього класу подібний дескриптор може не вміщувати лінгвістичну інформаційну складову. Як було сказано вище, екземпляр фонеми третього класу можна умовно поділити на дві складові – «паузу» і «сплеск». «Пауза» не несе жодної лінгвістичної інформації (хоча може бути використана як передумова). Якщо довжина сигналу значно перевищує 50 мс, буде згенеровано декілька дескрипторів для «паузи». Такі дескриптори не будуть репрезентативними, що знизить якість розпізнавання фонем даного класу. Тому для генерації навчальних даних з фонем третього класу було вирішено вирівняти їх з кінця до 50 мс і згенерувати один дескриптор саме для частини фонеми зі «сплеском».

Крім того, для підвищення якості даних для навчання, було вирішено для «стабільних в часі» фонем, якщо їх довжина перевищує 75 мс, відкинути зліва і справа сигналу фрагменти по 12,5 мс, що з великою ймовірністю є закінченням попередньої фонеми і початком наступної. Для «нестабільних в часі» фонем другого класу було вирішено відкинути 25 мс сигналу зліва і справа (відповідно до проведеного

попередньо аналізу бази даних). Такий підхід дозволяє збільшити ймовірність того, що система буде навчатись на даних, згенерованих із «чистого» звучання окремих фонем. За критерій оцінки побудованого дескриптора і класифікації фонем було взято результати навчання LSVM.

Розпізнавання мовленнєвого сигналу

Задачу розпізнавання окремих фонем можна ототожнити із задачею класифікації. В даній роботі використовувався LSVM для оцінки оптимальності побудови дескриптора і розбиття фонем на класи. Оскільки не існує загального підходу до вибору ядра для нелінійного SVM було вирішено відмовитися від даного підходу (при чисельних спробах підбору параметрів нелінійного SVM спостерігалося тільки погіршення якості розпізнавання). Для отримання результатів тренування LSVM було використано реалізацію SVMlight [7].

Як альтернативу при проектуванні дескриптора було розглянуто його варіанти, що включають один, два і чотири вікна з MFCC, а також їх розширення за допомогою MEDC (Delta and Acceleration Coefficients). Але варіант дескриптора, представлений вище, показав найкращі результати.

На рис. 11 представлені результати тренування LSVM у відповідності до проведеної класифікації фонем. В першому рядку наведені результати розбиття всіх фонем на два класи (перший це «стабільні в часі фонеми», другий – «нестабільні в часі» першого і другого типу). В другому рядку представлені результати розбиття кожного із верхніх класів ще на два. Для «стабільних в часі» – це голосні і приголосні, для другого – це «нестабільні в часі» першого і другого типів. Високий відсоток розпізнавання свідчить про лінійну роздільність простору фонем на визначені класи, що підтверджує коректність застосованого підходу.

Всі фонеми			
Стабільні в часі **(98.70%)**		Нестабільні в часі **(93.37%)**	
ay ae a aa i ii j y yy e ee oo u uu ur **(82.48%)**	n nn l ll m mm v vv d dd g gg b bb r rr **(83.48%)**	s c ss z zz ch sh sch zh h hh f ff **(98.57%)**	k kk p pp t tt **(85.40%)**

Рис. 11. Результати тренування LSVM (в дужках наведено відсоток правильно розпізнаних екземплярів (дескрипторів), на яких велось навчання)

Варто відмітити, що SVM (як і AdaBoost) є бінарним класифікатором, тобто виконується класифікація на два класи (простір розділяється на дві частини гіперплощиною). У випадку, коли необхідно виконати розпізнавання більше двох класів об'єктів, використовують декілька бінарних класифікаторів. У загальному випадку система повинна розпізнати кожну із 50-и фонем, тобто маємо 50 бінарних класифікаторів (звичайно, можна використати деревовидну структуру із бінарних класифікаторів і значно зменшити їх кількість, але оскільки кожен із класифікаторів має деяку ймовірність похибки, то така структура призведе до її накопичення). Така велика кількість бінарних класифікаторів негативно вплине на швидкодію системи і є збитковою, тому було вирішено об'єднати фонеми в групи, які і будуть тими мовленнєвими одиницями, що розпізнаються системою. Логічним є об'єднання дуже схожих фонем за своїм сенсом і звучанням, наприклад, в один клас можна об'єднати тверді і пом'якшені варіанти одного і того ж приголосного звуку, наголошені і ненаголошені варіанти голосної літери. Фінальне розділення класів представлене у табл. 1.

Таблиця 1

Мовленнєві одиниці для розпізнавання

Літера	Мовленнєва одиниця	Літера	Мовленнєва одиниця	Літера	Мовленнєва одиниця	Літера	Мовленнєва одиниця
i	i ii j (и и' й)	n	n nn (н н')	b	b bb (б б')	zh	zh (ж)
a	a aa ay (а а' ау)	l	l ll (л л')	r	r rr (р р')	h	h hh (х х')
y	y yy (ы ы')	m	m mm (м м')	s	s c ss (с ц с')	f	f ff (ф ф')
e	e ee ae (э э' аэ)	v	v vv (в в')	z	z zz (з з')	k	k kk (к к')
o	oo (о')	d	d dd (д д')	ch	ch (ч)	p	p pp (п п')
u	u uu ur (у у' ур)	g	g gg (г г')	sh	sh sch (ш щ)	t	t tt (т т')
pau	Пауза (мовленнєві одиниці відсутні)						

Отже, всього необхідно використати 25 бінарних класифікаторів, кожен з яких виділяє екземпляри однієї мовленнєвої одиниці з простору, що складається із всіх екземплярів. Висока ентропія природних мов дозволяє використати ці класи для розпізнавання окремих слів, навіть якщо буде допущена незначна похибка при класифікації.

Для навчання бінарного класифікатора необхідно надати дві вибірки даних. Перша вибірка включає так звані «позитивні» приклади (екземпляри) – дескриптори цільової мовленнєвої одиниці. Друга вибірка містить «негативні» приклади – дескриптори всіх інших мовленнєвих одиниць, що розпізнаються. На рис. 12 представлені результати тренування 25 бінарних LSVM класифікаторів. Як бачимо, велика кількість мовленнєвих одиниць взагалі не розпізнається за допомогою LSVM – навіть на даних, на яких велось тренування, відсоток розпізнаних «позитивних» екземплярів дорівнює нулю. Це означає, що неможливо

провести оптимальну гіперплощину, що виділить підпростір цільових мовленнєвих одиниць від всіх інших. Таким чином, LSVM не підходить для розпізнавання окремих мовленнєвих одиниць, хоча може бути успішно застосований для розпізнавання пауз, або класів фонем («стабільні в часі» і «нестабільні в часі»).

Отримані дані демонструють високу ступінь подібності між класами, що розпізнаються, оскільки неможливо ефективно виділити підкласи цільових мовленнєвих одиниць за допомогою гіперплощини. Отже, необхідно застосувати більш потужний класифікатор, який дозволить ефективно вирішити цю задачу. За такий класифікатор було вирішено взяти AdaBoost на основі дерева рішень.

pau	i	a	y	e	o	u	n	l	m	v	d	g	b	r	s	z	ch	sh	zh	h	f	k	p	t
99	37	71	0	0	33	0	59	39	43	0	0	0	5	0	95	73	63	62	3	0	0	0	0	0

Рис. 12. Результати тренування LSVM для кожної із мовленнєвих одиниць (перший рядок – умовні позначення мовленнєвих одиниць, другий – відсоток правильно класифікованих екземплярів по відношенню до вибірки саме «позитивних» екземплярів)

В даній роботі для побудови моделі використовувалась реалізація AdaBoost Alexander Vezhnevets [8]. Всі наступні результати були отримані з використанням Gentle AdaBoost (далі – AdaBoost).

Для оцінки результатів розпізнавання, була запропонована наступна методика. Оскільки експерименти проводилися на досить невеликій базі даних, було вирішено проводити навчання з вилученням одного із файлів. По завершенню навчання цей файл надходив на вхід системи для розпізнавання мовленнєвих одиниць. Далі із навчальної вибірки вилучався наступний файл і процес повторювався. Для отримання статистики розпізнавання кожної із мовленнєвих одиниць, над вхідним файлом проводилася сегментація (файл містить по-фонемну розмітку, тому можливо отримати відоме очікуване значення на виході системи по завершенню розпізнавання), після чого відбувалося порівняння із отриманим результатом і очікуваним. В такому випадку можливі три ситуації: якщо значення співпало, то одиниця розпізнана коректно, якщо ні – одиниця розпізнана некоректно або взагалі не розпізнана. Приклад результату роботи системи показаний на рис. 13.

```
expected 9 (n)
   - 0 (*)
   - 9 (n)
   - 9 (n)
   - 0 (*)
  - 11 (m)
```

Рис. 13. Приклад даних на виході системи

Оскільки в загальному випадку довжина окремої фонеми не є статичною, маємо декілька спрацювань. В ідеальному випадку повинні бути отримані 5 розпізнаних «n». Але, як бачимо, було отримано два нерозпізнаних вікна (*), два вірно розпізнаних (n) і одне хибне спрацювання (m). Процес розпізнавання зводиться до генерації дескриптора для кожного вікна в 50 мс, далі цей дескриптор надходить до 25 бінарних класифікаторів – отримуємо 25 значень, з яких обираємо максимальне, що і відповідає розпізнаній мовленнєвій одиниці. Якщо максимальне значення менше нуля, то дескриптор вважається не розпізнаним.

Результати зі статистикою розпізнавання кожної з мовленнєвих одиниць представлені на рис.14.

Якщо значення в стовбці вірно розпізнаних значно перевищує значення в стовбці хибно розпізнаних, і більше або рівне одиниці, то можливо застосовувати правило максимуму для чіткої ідентифікації мовленнєвої одиниці. Як бачимо з представлених результатів, такі мовленнєві одиниці, як «k», «p», «y» і «t», досить погано розпізнаються – число хибних спрацювань перевищує число вірно розпізнаних. Крім того, система характеризується високим показником вірно розпізнаних пауз, що дозволяє підвищити швидкодію, виконуючи аналіз дескриптора спочатку на наявність «паузи»: якщо отримано позитивну відповідь (більше 0), то виконувати перевірку інших класифікаторів немає сенсу. Наведені результати є досить умовними, через те, що хибно розпізнані мовленнєві одиниці, як правило, дуже схожі за звучанням до цільової, і при побудові, наприклад, системи розпізнавання команд, не будуть значно впливати на якість розпізнавання (цьому також сприяє висока ентропія природних мов). Але опираючись на дані результати можна робити оцінку ефективності прийнятих рішень при побудові системи розпізнавання. Високі значення хибно розпізнаних і не розпізнаних мовленнєвих одиниць на екземпляр засвідчують необхідність додаткової обробки результатів і ускладнення системи.

Евристична обробка результатів розпізнавання

При аналізі результатів класифікації окремої мовленнєвої одиниці (вектор із 24 значень) було помічено, що якщо задатися певним параметром відстані і брати до уваги відразу декілька максимальних значень, які не перевищують даний параметр (назвемо його threshold), отримуємо додаткові гіпотези щодо класифікації сигналу. При threshold = 5 вихід системи представлений на рис. 15.

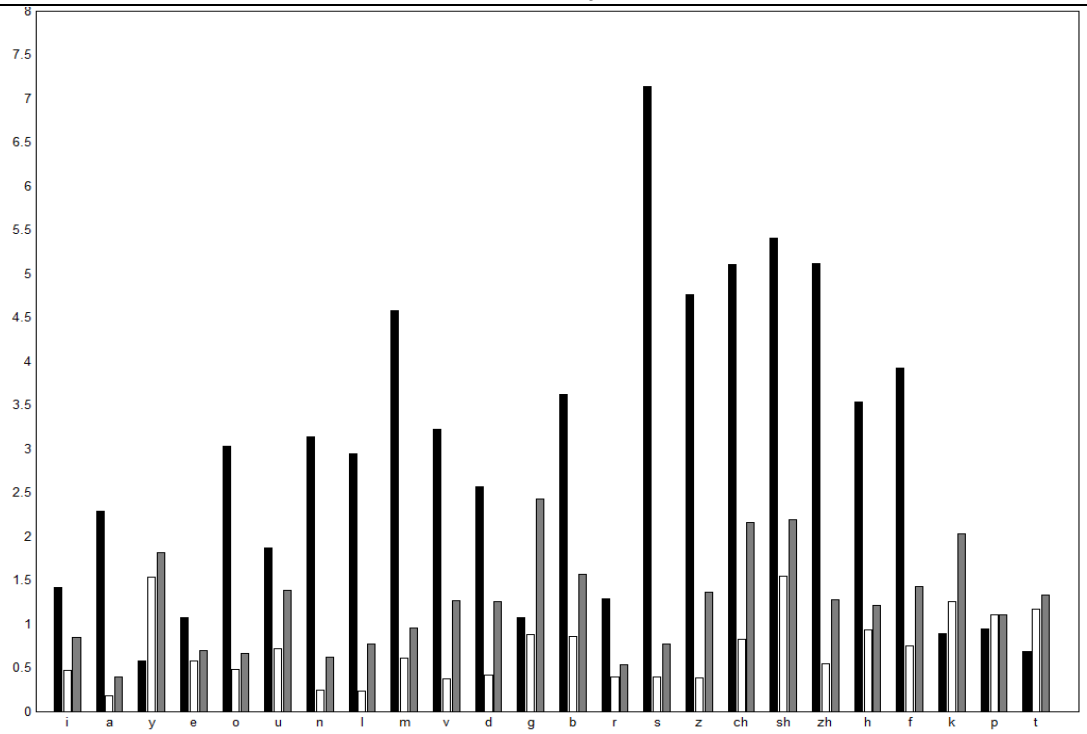

Рис. 14. Результати розпізнавання мовленнєвих одиниць без додаткової обробки. Крок вікна сканування 12,5 мс. Чорний колір – кількість вірно розпізнаних мовленнєвих одиниць на екземпляр, білий – кількість хибно розпізнаних, сірий – кількість нерозпізнаних мовленнєвих одиниць на екземпляр

```
expected 9 (n)
   - 0 (*) (0) n (-4.471661)
   - 9 (n) (1.648131)
   - 9 (n) (2.731240)  m (-0.770233)
   - 0 (*) (0)  n (-1.146550) m (-1.200526)
   - 11 (m) (6.850019)
```

Рис. 15. Приклад даних на виході системи при threshold = 5

Як бачимо, якщо додати певні правила вибору вірної гіпотези, можливо підвищити кількість вірно розпізнаних мовленнєвих одиниць (в наведеному прикладі при виборі замість «*» наступної гіпотези отримуємо правильну відповідь).

Якщо виконати аналіз деякого тексту, який має певне змістове навантаження, то можна виявити закономірності в слідуванні букв одна за одною. Тобто можна побудувати таблицю ймовірностей слідування мовленнєвих одиниць (рис. 16).

Як бачимо, розподілення ймовірності є неоднорідним. Це означає, що можливо вводити певну корекцію у розпізнавання мовленнєвих одиниць, враховуючи поточну розпізнану одиницю і гіпотези по класифікації цільової мовленнєвої одиниці.

В ході даної роботи запропоновано використати наступний евристичний підхід. Нехай маємо декілька гіпотез відсортованих по мірі їх правдоподібності за результатами AdaBoost (табл. 2). Тоді найбільш ймовірна гіпотеза буде визначатися як:

$$\max_i \left((\alpha - \beta N_i) P_i + (\gamma k | F_0 = F_i) \right), \quad (1)$$

де N_i – номер гіпотези мовленнєвої одиниці;

F_i – умовне позначення (або код) мовленнєвої одиниці;

R_i – значення в масиві класифікації за допомогою AdaBoost;

P_i – ймовірність слідування мовленнєвої одиниці, якій відповідає дана гіпотеза, за уже розпізнаною мовленнєвою одиницею; вже розпізнану мовленнєву одиницю позначимо як F_0;

α, β, γ – деякі константи;

k – кількість безперервно розпізнаних однакових мовленнєвих одиниць.

	i	a	y	e	o	u	n	l	m	v	d	g	b	r	s	z	ch	sh	zh	h	f	k	p	t
i	0.156	0.048	0.000	0.028	0.030	0.007	0.084	0.073	0.063	0.070	0.038	0.018	0.020	0.023	0.099	0.047	0.036	0.010	0.008	0.032	0.003	0.051	0.036	0.073
a	0.093	0.040	0.000	0.015	0.016	0.008	0.097	0.121	0.051	0.064	0.036	0.017	0.019	0.045	0.076	0.058	0.017	0.022	0.020	0.015	0.008	0.077	0.034	0.082
y	0.204	0.007	0.000	0.091	0.020	0.007	0.043	0.115	0.103	0.095	0.018	0.010	0.029	0.022	0.073	0.012	0.017	0.026	0.004	0.076	0.002	0.026	0.034	0.055
e	0.016	0.000	0.000	0.015	0.000	0.002	0.028	0.033	0.016	0.016	0.006	0.005	0.001	0.012	0.012	0.003	0.000	0.001	0.000	0.002	0.008	0.055	0.005	0.779
o	0.099	0.012	0.000	0.044	0.020	0.006	0.083	0.070	0.060	0.100	0.057	0.056	0.048	0.069	0.095	0.018	0.023	0.012	0.023	0.006	0.005	0.028	0.034	0.079
u	0.055	0.015	0.000	0.008	0.015	0.005	0.047	0.051	0.054	0.054	0.080	0.057	0.032	0.039	0.074	0.033	0.053	0.045	0.054	0.014	0.002	0.050	0.051	0.069
n	0.346	0.220	0.066	0.001	0.178	0.036	0.062	0.001	0.002	0.007	0.017	0.006	0.004	0.003	0.030	0.003	0.004	0.004	0.001	0.001	0.002	0.011	0.008	0.022
l	0.341	0.189	0.020	0.003	0.155	0.030	0.065	0.008	0.004	0.014	0.012	0.007	0.006	0.005	0.045	0.007	0.013	0.012	0.009	0.001	0.001	0.042	0.018	0.010
m	0.297	0.132	0.044	0.005	0.158	0.086	0.063	0.011	0.014	0.026	0.013	0.010	0.011	0.008	0.037	0.005	0.017	0.002	0.004	0.002	0.002	0.021	0.037	0.011
v	0.214	0.154	0.064	0.008	0.190	0.024	0.044	0.033	0.010	0.011	0.015	0.008	0.005	0.025	0.093	0.014	0.005	0.025	0.001	0.002	0.001	0.019	0.023	0.019
d	0.303	0.182	0.023	0.002	0.141	0.062	0.069	0.032	0.007	0.038	0.004	0.002	0.004	0.067	0.028	0.002	0.003	0.004	0.005	0.002	0.001	0.010	0.010	0.010
g	0.106	0.069	0.000	0.001	0.502	0.039	0.023	0.080	0.003	0.006	0.068	0.001	0.002	0.071	0.009	0.002	0.002	0.000	0.000	0.000	0.000	0.007	0.006	0.002
b	0.231	0.096	0.242	0.004	0.165	0.068	0.026	0.058	0.003	0.005	0.001	0.001	0.002	0.077	0.012	0.001	0.001	0.021	0.001	0.007	0.000	0.012	0.001	0.000
r	0.337	0.241	0.043	0.001	0.180	0.070	0.022	0.003	0.012	0.013	0.010	0.010	0.006	0.003	0.012	0.002	0.003	0.008	0.010	0.003	0.001	0.010	0.007	0.017
s	0.183	0.102	0.007	0.002	0.076	0.023	0.036	0.057	0.025	0.044	0.010	0.003	0.005	0.009	0.032	0.003	0.011	0.003	0.002	0.005	0.001	0.116	0.047	0.264
z	0.085	0.358	0.047	0.004	0.051	0.029	0.127	0.017	0.046	0.064	0.052	0.031	0.015	0.022	0.020	0.007	0.002	0.002	0.009	0.000	0.001	0.016	0.012	0.005
ch	0.417	0.154	0.000	0.000	0.006	0.042	0.063	0.004	0.001	0.003	0.001	0.000	0.001	0.002	0.002	0.000	0.001	0.007	0.000	0.000	0.000	0.019	0.002	0.272
sh	0.538	0.172	0.000	0.001	0.036	0.039	0.050	0.056	0.005	0.005	0.002	0.001	0.001	0.003	0.008	0.001	0.003	0.001	0.000	0.000	0.000	0.049	0.008	0.018
zh	0.564	0.128	0.000	0.001	0.011	0.019	0.135	0.001	0.002	0.002	0.104	0.002	0.006	0.001	0.003	0.001	0.005	0.000	0.002	0.000	0.000	0.009	0.002	0.002
h	0.090	0.108	0.000	0.009	0.298	0.028	0.051	0.031	0.028	0.056	0.024	0.017	0.018	0.030	0.068	0.011	0.011	0.003	0.006	0.002	0.006	0.034	0.059	0.020
f	0.388	0.096	0.006	0.003	0.134	0.069	0.006	0.033	0.007	0.007	0.001	0.002	0.003	0.188	0.018	0.002	0.001	0.002	0.000	0.000	0.017	0.005	0.006	0.007
k	0.125	0.247	0.000	0.004	0.301	0.049	0.055	0.018	0.006	0.023	0.007	0.004	0.011	0.051	0.026	0.004	0.004	0.001	0.008	0.001	0.001	0.012	0.012	0.033
p	0.179	0.065	0.014	0.000	0.372	0.028	0.005	0.029	0.000	0.000	0.000	0.000	0.000	0.299	0.006	0.000	0.003	0.000	0.000	0.000	0.001	0.003	0.004	0.004
t	0.212	0.123	0.025	0.005	0.280	0.032	0.039	0.008	0.007	0.073	0.011	0.004	0.007	0.055	0.058	0.004	0.012	0.001	0.002	0.002	0.001	0.017	0.017	0.011

Рис. 16. Таблиця ймовірностей слідування цільових мовленнєвих одиниць, побудована на ймовірностях слідування букв для декількох текстів різних авторів

Таблиця 2

Таблиця відсортованих за ймовірністю гіпотез на виході системи

$N_1 =$ 1	F_1	R_1	P_1
$N_2 =$ 2	F_2	R_2	P_2
$N_3 =$ 3	F_3	R_3	P_3
…	…	…	…

Також необхідно враховувати наступні правила:

1. Якщо перша гіпотеза відповідає нерозпізнаній мовленнєвій одиниці (*), то параметр threshold потрібно брати меншим, ніж заданий (наприклад, threshold = 1).

2. Якщо мовленнєва одиниця, що відповідає першій гіпотезі, дорівнює уже розпізнаній попередній одиниці, і не дорівнює *, то за найбільш ймовірну слід обрати першу гіпотезу.

Аналіз результатів

Результати із застосуванням запропонованої евристики були отримані при наступних параметрах: α = 1.5, β = 0.1, γ = 0.05, threshold = 5, threshold* = 1, максимальна кількість гіпотез – 4. За таблицю з ймовірностями слідування мовленнєвих одиниць була взята таблиця на рис. 16. Також представлені результати з різним кроком вікна сканування (рис. 17, рис. 18).

В загальному випадку застосування такого підходу дозволило збільшити кількість вірно розпізнаних мовленнєвих одиниць на екземпляр. Збільшення кількості невірно розпізнаних мовленнєвих одиниць пояснюється застосуванням таблиці слідування саме букв, які були ототожнені з мовленнєвими одиницями, що розпізнаються. Слід зазначити, що у більшості людських мов існує деяка невідповідність між написанням і вимовлянням слів, що вносить деяку похибку у таблицю ймовірностей. Також потрібно виконати пошук оптимальних параметрів алгоритму (α, β, γ, threshold, threshold*). Але, навіть при не зовсім коректній таблиці ймовірностей, такий евристичний підхід показує поліпшення результатів (кількість вірно розпізнаних збільшується, кількість нерозпізнаних зменшується). Крім того, запропонований підхід дозволяє отримати на виході системи, як найбільш ймовірні по розрахункам гіпотези, так і всі інші, що може слугувати як додаткова інформація для розпізнавання слів по словнику.

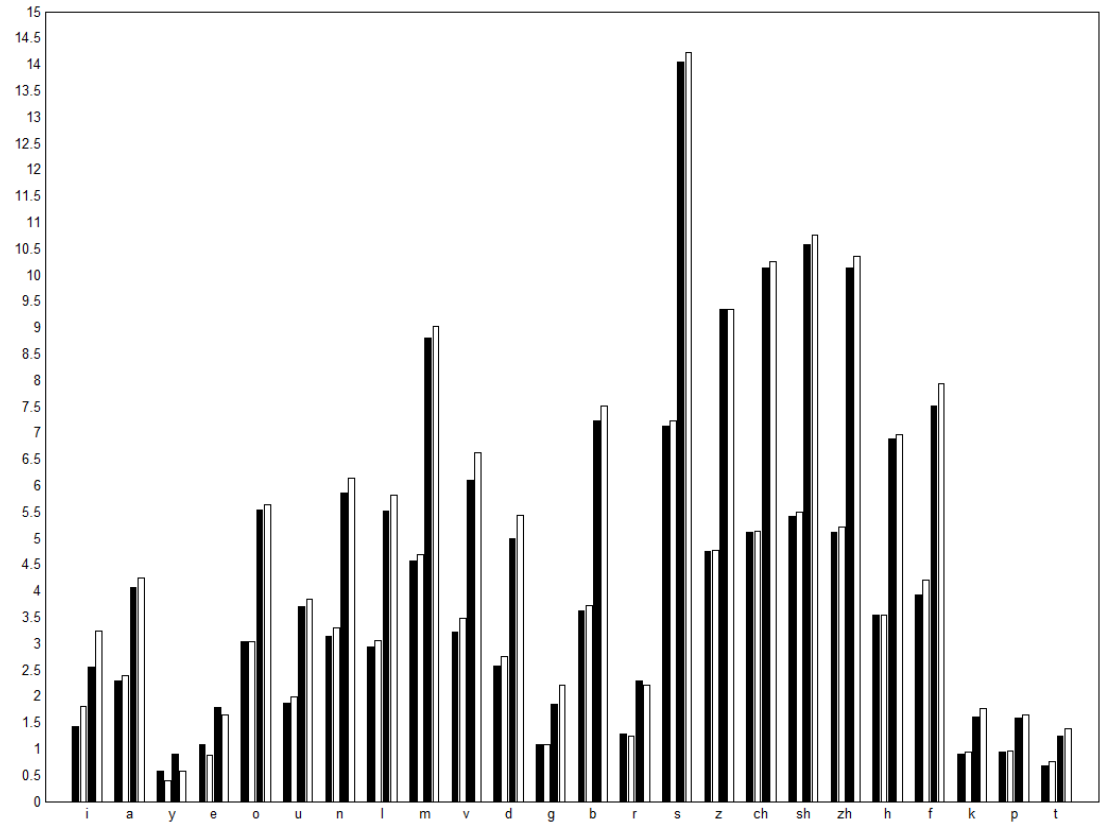

Рис. 17. Порівняння кількості вірно розпізнаних мовленнєвих одиниць на екземпляр для 4-х випадків: перший стовпчик чорного кольору – крок вікна сканування 12.5 мс, без евристики; другий стовпчик білого кольору – крок вікна сканування 12.5 мс, евристика; третій стовпчик чорного кольору – крок вікна сканування 6.25 мс, без евристики; четвертий стовпчик білого кольору – крок вікна сканування 6.25 мс, евристика

При побудові системи з деякою скінченною і обмеженою кількістю слів (наприклад, розпізнавання команд) можливо застосувати спеціальний файл із записаною транскрипцією команд у мовленнєвих одиницях, по якому і побудувати таблицю ймовірностей переходів між мовленнєвими одиницями, що підвищить якість розпізнавання.

Загальну оцінку побудованій системі розпізнавання можна дати виходячи з результатів на рис. 18. Чим більше значення відношення вірно розпізнаних до хибно розпізнаних мовленнєвих одиниць на екземпляр, тим більша ймовірність правильного розпізнавання мовленнєвої одиниці. Крім того, якщо дане відношення значно перевищує 1, то можливо ввести додаткове правило, за яким мовленнєва одиниця вважається розпізнаною, якщо кілька вікон сканування підряд дають однаковий результат.

Зменшення кроку вікна сканування дозволяє підвищити якість розпізнавання. Застосування евристики в загальному випадку хоч і дозволяє збільшити кількість вірно розпізнаних мовленнєвих одиниць, але разом з тим значно збільшується і кількість хибно розпізнаних, що засвідчує необхідність пошуку оптимальних параметрів евристики і більш точної таблиці переходів мовленнєвих одиниць.

Також слід відмітити, що значний позитивний вплив на якість розпізнавання мовленнєвих одиниць чинять такі параметри навчання AdaBoost, як кількість ітерацій навчання і максимальна кількість дочірніх вузлів на вузол дерева рішень. На рис. 19 представлені дані розпізнавання для кількості ітерацій навчання 150 і максимальній кількості дочірніх вузлів 8.

Висновки

Високі показники відношення вірно розпізнаних мовленнєвих одиниць до хибно розпізнаних для класів вказують на ефективність розробленої системи. При відповідному тренуванні на зразках вимови слів конкретним диктором систему можливо використовувати для розпізнавання сталого набору слів, наприклад, мовленнєвих команд.

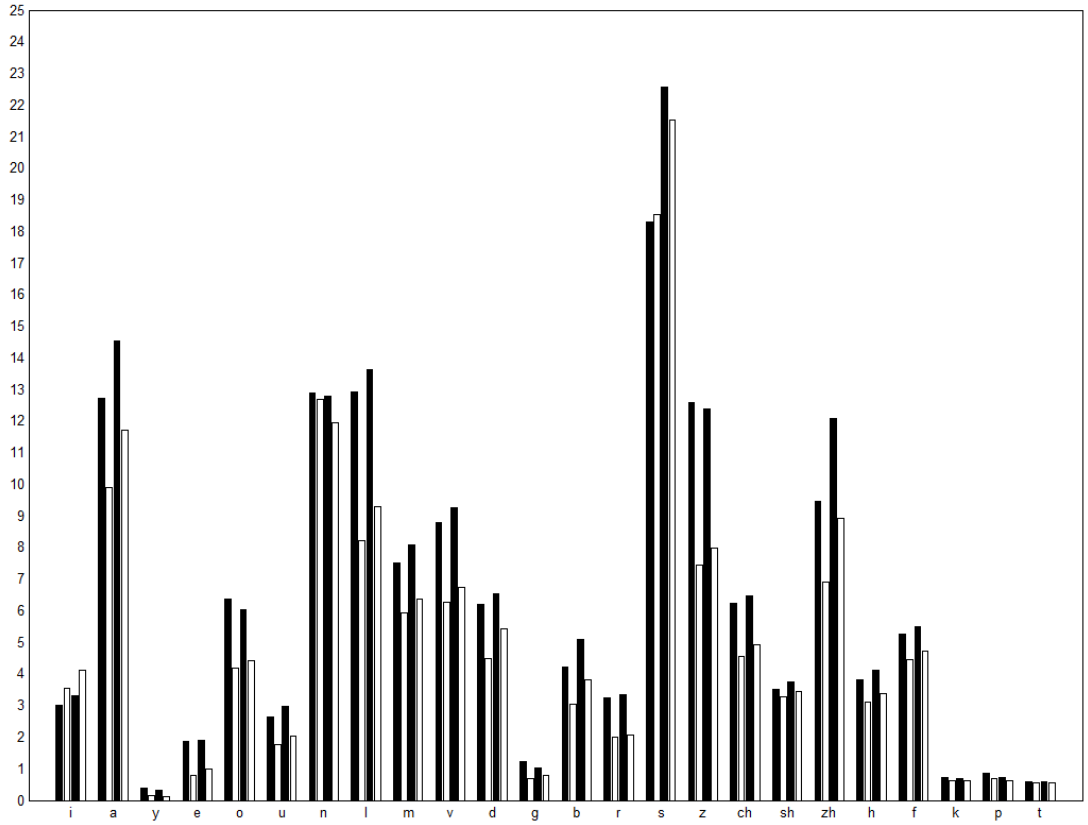

Рис. 18. Відношення вірно розпізнаних до хибно розпізнаних мовленнєвих одиниць на екземпляр для 4-х випадків

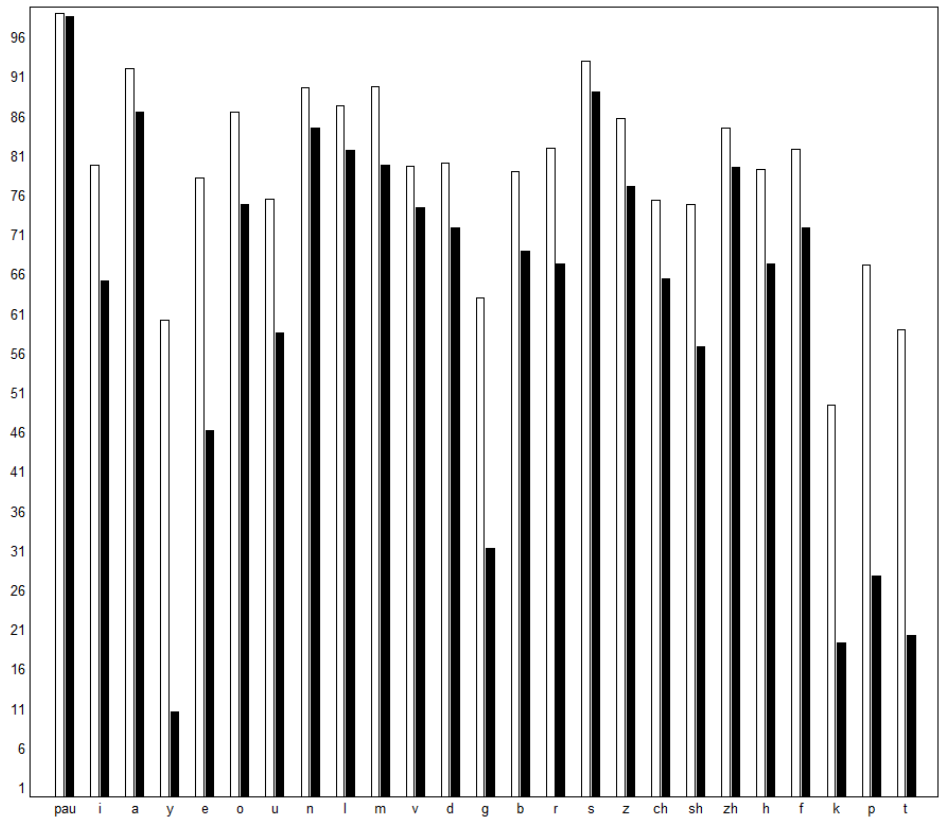

Рис. 19. Відсоток вірно розпізнаних мовленнєвих одиниць (чорний колір) і вірно розпізнаних+нерозпізнаних (білий колір) по відношенню до всіх мовленнєвих одиниць, що очікуються на виході системи

Запропонований метод розпізнавання мови на основі AdaBoost дозволяє досягти вірного розпізнавання 63–78% мовленнєвих одиниць, що є порівняним з характеристиками відомих існуючих систем [9]. Крім того, система дозволяє отримати на виході додатково інформацію по всім гіпотезам відносно кожної із мовленнєвих одиниць, а також має додатковий стан, який відповідає сигналу, що не вдалося розпізнати. Це дозволяє виконувати покращення якості розпізнавання доданням нових алгоритмів і

методик із застосуванням статистичних даних, різного роду словників, особливостей мови. Покращення часових показників системи можливо шляхом розпаралелювання обчислень [10].

Література

1. Furui, Sadaoki. 50 years of Progress in speech and Speaker Recognition Research / Sadaoki Furui // ECTI Transactions on Computer and Information Technology. – Vol. 1, No. 2. – November 2005.

2. Mermelstein, P. Distance measures for speech recognition, psychological and instrumental / P. Mermelstein ; C. H. Chen, Ed. – Pattern Recognition and Artificial Intelligence. – Academic, New York, 1976. – pp. 374-388.

3. Stevens, S. The relation of pitch to frequency / S. Stevens // American Journal of Psychology. – vol. 53(3). – 1940. – pp. 329-353.

4. Benesty, J. Speech Enhancement / J. Benesty, S. Makino, J. Chen. – Springer, 2005. – ISBN 978-3-540-24039-6.

5. Шмирев Н. Акустическая база данных / Н. В. Шмырев. – Copyright (c), 2005.

6. Begam, M. Voice Recognition Algorithms using Mel Frequency Cepstral Coefficient and Dynamic Time Warping Techniques / Lindasalwa Muda, Mumtaj Begam and I. Elamvazuthi // Journal of Computing. – vol. 2, issue 3. – March 2010. – ISSN 2151-9617.

7. SVMlight [Електронний ресурс]. – Режим доступу : http://svmlight.joachims.org/

8. GML AdaBoost Toolbox by Alexander Vezhnevets [Електронний ресурс]. – Режим доступу : http://www.inf.ethz.ch/personal/vezhneva/#code

9. Ладошко О. Исследование влияния характеристик телефонного канала связи на надёжность распознавания фонем / О.М. Ладошко // Международная научно-техническая конференция студентов, аспирантов и молодых ученых. Информационные управляющие системы и компьютерный мониторинг (ИУС и КМ-2012), Сб. трудов. – Донецк, 2012. – С. 143–148.

10. Павловець, О. Система розпізнавання мовленнєвих одиниць на основі ADABOOST з прискоренням на паралельних архітектурах / Є.С. Сулема, О.В. Павловець // V конференція молодих вчених "Прикладна математика та комп'ютинг" (ПМК-2013), збірник тез доповідей. – К., 2013. – С. 329–334.

References

1. Sadaoki Furui, "50 years of Progress in speech and Speaker Recognition Research" in ECTI Transactions on Computer and Information Technology, Vol. 1, No. 2, November 2005.

2. P. Mermelstein, "Distance measures for speech recognition, psychological and instrumental" in Pattern Recognition and Artificial Intelligence, C. H. Chen, Ed., Academic, New York, 1976, pp. 374–388.

3. S. Stevens, "The relation of pitch to frequency", American Journal of Psychology, vol. 53(3), 1940, pp. 329-353.

4. J. Benesty, S. Makino, J. Chen, "Speech Enhancement", Springer, 2005, ISBN 978-3-540-24039-6.

5. Nickolay V. Shmyrev, "Acoustic Database" [File]. – Copyright (c) 2005.

6. Lindasalwa Muda, Mumtaj Begam and I. Elamvazuthi, "Voice Recognition Algorithms using Mel Frequency Cepstral Coefficient and Dynamic Time Warping Techniques" in Journal of Computing, volume 2, issue 3, March 2010, ISSN 2151-9617.

7. "SVMlight" (04-15-2013), http://svmlight.joachims.org/

8. "GML AdaBoost Toolbox by Alexander Vezhnevets" (04-15-2013), http://www.inf.ethz.ch/personal/vezhneva/#code

9. Ladoshko O. M., "Issledovanie vlijanija kharakteristik telefonnogo kanala svjazi na nadjozhnost' raspoznavanija fonem" in Mezhdunarodnaya nauchno-tekhnicheskaya konferenciya studentov, aspirantov i molodykh uchenykh. Informacionnye upravlyayushhie sistemy i kompyuternyy monitoring, Donetsk, 2012, pp. 143–148.

10. Yevgeniya Sulema, Oleksandr Pavlovets, "Systema rozpiznamannya movlennevyh odynyts na osnovi ADABOOST z pryskorennyam na paralelnyh arhitekturah", the 5th Conference of young scientists "Prykladna matematyka ta kompyutyng" (PMK-2013), Kyiv, 2013, pp. 329–334.

Рецензія/Peer review : 24.4.2013 р. Надрукована/Printed :19.6.2013 р.
Рецензент: д.т.н., проф. Дичка І.А., завідувач кафедри програмного забезпечення комп'ютерних систем Національного технічного університету України «Київський політехнічний інститут»