Міністерство освіти і науки України
Харківський національний університет імені В. Н. Каразіна
Навчально-науковий інститут екології, зеленої енергетики та сталого розвитку
Кафедра інформаційних технологій у відновлюваній енергетиці

До захисту допущено
Кафедрою інформаційних технологій у відновлюваній енергетиці протокол №____ від _____

в.о. завідувача кафедри _____ *Ілля МАРУЩЕНКО*
(підпис) *(ім'я, прізвище)*

«___»_____ 2025 р.

Кваліфікаційна робота
здобувача Антона КОРОБКИ другого (магістерського) рівня вищої освіти

Штучний інтелект у задачах розпізнавання голосових команд
(назва роботи)

Спеціальність (спеціалізація) 105 Прикладна фізика та наноматеріали
(код та найменування спеціальності; спеціалізації спеціальності - за наявності)
Освітня програма Інформаційні технології в енергетичних системах
(назва освітньої програми)

Виконавець _____ Антон КОРОБКА
(підпис) *(ім'я, прізвище)*

Науковий керівник _____ *Руслан СУХОВ*
(підпис) *(ім'я, прізвище)*

Харків – 2025

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

Харківський національний університет імені В. Н. Каразіна

Навчально-науковий інститут екології, зеленої енергетики та сталого розвитку

Кафедра інформаційних технологій у відновлюваній енергетиці

Рівень вищої освіти (освітньо-кваліфікаційний рівень) магістр

Спеціальність 105 «Прикладна фізика та наноматеріали»

ЗАТВЕРДЖУЮ

В.о. завідувача кафедри

_____ Ілля МАРУЩЕНКО

(підпис)

“_____” _____ 2025 року

ЗАВДАННЯ

НА КВАЛІФІКАЦІЙНУ РОБОТУ

Коробці Антону Віталійовичу

1. Тема роботи «Штучний інтелект у задачах розпізнавання голосових команд», керівник роботи доцент Сухов Руслан Володимирович, затверджено наказом по університету від 25 листопада 2025 року №3401-5/4196.
2. Строк подання студентом роботи 01.12.2025 р.
3. Перелік питань, які потрібно розробити:

   1. Провести аналітичний огляд методів цифрової обробки аудіосигналів та алгоритмів розпізнавання мовлення.

   2. Дослідити перцептивні підходи до представлення мовних сигналів (мел-шкала, мел-кепстральні коефіцієнти MFCC) і їхню роль у підвищенні точності розпізнавання.

   3. Розробити програмно-апаратну архітектуру пристрою на ESP32-S3: схему збирання аудіоданих, попередню обробку сигналу (фільтрація шумів, нормалізація, детекція голосової активності) та інтеграцію нейронної

мережі для розпізнавання команди.

4. Провести експериментальне тестування прототипу, оцінити точність розпізнавання в різних умовах та проаналізувати продуктивність системи.

5. Інтегрувати в існуючу систему автоматизації, яка використовується для проведення фізичних експериментів з нанорозмірними плівковими системами.

4. План роботи:

№ з/п	Назви етапів роботи
1	Вибір теми
2	Затвердження теми на кафедрі
3	Робота над літературою та іншими джерелами
4	Написання вступу
5	Написання першого розділу
6	Написання другого розділу
7	Проведення експерименту
8	Аналіз результатів експерименту
9	Завершення написання дипломної роботи (висновки та література)
10	Подача науковому керівнику
11	Відгук, рецензія, підготовка до захисту
12	Попередній захист
13	Захист

5. Дата видачі завдання: 03 листопада 2025 року.

Студент _____ Антон КОРОБКА
підпис

Керівник роботи _____ Руслан СУХОВ
підпис

РЕФЕРАТ

Коробка Антон. «Штучний інтелект у задачах розпізнавання голосових команд».

Кваліфікаційна робота магістра з прикладної фізики. – Харківський національний університет імені В. Н. Каразіна - 49 с.

Кваліфікаційна робота складається з вступу, двох розділів, висновку та переліку використаних джерел.

Робота містить 34 графічних зображень, 31 використаних літературних джерел.

Мета дослідження: розробка програмно-апаратного комплексу для розпізнавання голосових команд.

Актуальність: Дипломна робота є актуальною, оскільки демонструє створення функціонального пристрою з розпізнаванням голосових команд на основі відкритих технологій, без використання хмарних сервісів. Це дозволяє забезпечити автономність, приватність і доступність рішення у медичних закладах, стерильних середовищах, системах розумного будинку, навчальних або демонстраційних стендах.

КЛЮЧОВІ СЛОВА: АУДІОСИГНАЛИ, FFT, СПЕКТОГРАМА, НЕЙРОМЕРЕЖА, AFE, МІКРОКОНТРОЛЕР.

ABSTRACT

Korobka Anton. “Artificial Intelligence in Voice Command Recognition Tasks.” Master’s Thesis in Applied Physics – V. N. Karazin Kharkiv National University – 49 pages.

The thesis consists of an introduction, two chapters, a conclusion, and a list of references.

The work contains 34 graphical figures and 31 bibliographic sources.

Research objective: development of a hardware–software system for voice command recognition.

Relevance: The thesis is relevant as it demonstrates the creation of a functional voice-recognition device based on open technologies without the use of cloud services. Such an approach ensures autonomy, privacy, and accessibility of the solution in medical facilities, sterile environments, smart-home systems, and educational or demonstration setups.

KEYWORDS: AUDIO SIGNALS, FFT, SPECTROGRAM, NEURAL NETWORK, AFE, MICROCONTROLLER.

ЗМІСТ

ВСТУП

Сучасний розвиток інформаційних технологій характеризується стрімким зростанням обсягів даних та підвищенням вимог до зручності взаємодії людини з технічними системами. Одним із найперспективніших напрямів у цій сфері є використання штучного інтелекту для обробки та інтерпретації мовлення. Голосове керування поступово стає невід'ємним елементом у побутових пристроях, промислових системах та в інтернеті речей (IoT), забезпечуючи більш природний і швидкий спосіб взаємодії користувача з машиною.

Задача розпізнавання голосових команд є багатокомпонентною і поєднує в собі методи цифрової обробки сигналів, машинного навчання та лінгвістичного аналізу. Класичні алгоритми, засновані на статистичних моделях, уже значною мірою поступаються сучасним нейронним мережам, які демонструють високу точність навіть у складних умовах — за наявності шумів, акцентів чи індивідуальних особливостей голосу.

Актуальність теми обумовлена широким використанням голосових інтерфейсів у системах «розумного дому», мобільних застосунках, медичних та освітніх технологіях. Застосування ШІ у цій галузі відкриває можливості не лише для підвищення точності розпізнавання, а й для адаптації систем до конкретного користувача, розширення функціональності та забезпечення багатомовної підтримки.

Метою даної роботи є розробка програмно-апаратного комплексу для розпізнавання голосових команд.

Для досягнення поставленої мети необхідно вирішити наступні завдання:

1. Виконати аналітичний огляд сучасного стану алгоритмів для розпізнавання голосу;

2. Дослідити перцептивні підходи до представлення мовних сигналів (мел-шкала, мел-кепстральні коефіцієнти MFCC) і їхню роль у підвищенні точності розпізнавання;
3. Створити апаратну частину пристрою для розпізнавання голосових команд;
4. Створити програмне забезпечення для мікроконтролера розробленого пристрою;
5. Розробити прототип програмно-апаратного комплексу, використовуючи сучасну елементну базу;
6. Перевірити розроблений пристрій в рамках автоматизації фізичного експерименту.

РОЗДІЛ ПЕРШИЙ

Обробка звуку

Звук є однією з основних форм передачі інформації у природі та технічних системах. Його аналіз, збереження та відтворення мають важливе значення в науці, техніці та повсякденному житті. Обробка звукових сигналів охоплює методи перетворення акустичних коливань у цифрову форму, їх подальший аналіз, фільтрацію, стиснення та синтез. Сучасні технології дозволяють не лише зберегти звук з високою якістю, але й виділяти з нього потрібні компоненти, розпізнавати мовлення, усувати шумові завади та створювати нові аудіоефекти.

У цьому розділі розглядаються основні принципи роботи із звуковими сигналами, їх математичні моделі та практичні аспекти застосування алгоритмів обробки звуку в сучасних пристроях і програмних комплексах.

Перед тим як аудіосигнал надходить до нейронної мережі для розпізнавання або аналізу, він проходить низку етапів обробки. Мета цих кроків – перетворити “сирий” звук у компактне й інформативне подання, яке зручно аналізувати алгоритмам машинного навчання.

1. Рівняння хвилі

На цьому рівні звук розглядається як фізичне явище — коливання тиску в середовищі, що поширюються у вигляді хвиль. Рівняння хвилі формалізує цю поведінку та описує, як енергія передається в просторі та часі. Це фундаментальна модель, з якої починається будь-яке математичне або технічне представлення звукового сигналу.

2. Розклад у ряд Фур’є

Будь-який складний звуковий сигнал можна уявити як суперпозицію синусоїд різної частоти, амплітуди та фази. Такий розклад дозволяє перейти

від хаотичних коливань до впорядкованої системи частотних компонент. Цей підхід є базою для всіх сучасних методів спектрального аналізу та дозволяє визначити, які частоти формують звук.

3. STFT / FFT

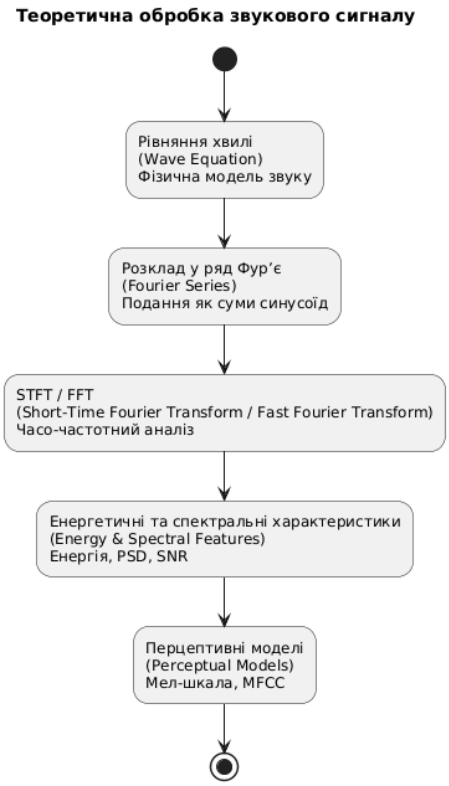
Методи швидкого перетворення Фур'є (FFT) та короткочасного перетворення Фур'є (STFT) потрібні, щоб побачити, як частотний склад сигналу змінюється з часом. Вони дозволяють перетворити часовий сигнал у частотну область і побудувати спектрограми. Це критично важливо для аналізу мови та шумів, оскільки саме в часо-частотному просторі можна відстежити фонеми, інтонації та завади.

4. Енергетичні та спектральні характеристики

На цьому рівні з сигналу витягуються числові ознаки, які описують його загальну форму: енергія, потужність, спектральна щільність та відношення сигнал/шум. Вони дозволяють кількісно оцінити, наскільки сильний чи чистий сигнал, виділити важливі частотні діапазони та відфільтрувати неінформативні компоненти. Такі параметри часто використовуються як попередні характеристики для подальшого аналізу.

5. Перцептивні моделі (мел-шкала, MFCC)

Людське вухо сприймає частоти нелінійно, тому виникає потреба у перцептивних моделях. Мел-шкала перетворює спектральну інформацію у форму, ближчу до реального слухового сприйняття. На її основі формуються мел-кепстральні коефіцієнти (MFCC) — компактне представлення звуку, яке добре описує тембр і мовні особливості. Цей етап забезпечує зв'язок між фізичним сигналом і його інтерпретацією в системах розпізнавання.



Діаграма обробки звукового сигналу

Рівняння хвилі

1. Перехід від неперервного сигналу до відліків

Звук у природі існує як неперервна функція часу s(t), де t змінюється безперервно, а значення функції — це миттєве відхилення тиску (або напруги з мікрофона).
Однак комп’ютери та мікроконтролери не можуть працювати з нескінченними неперервними функціями — вони оперують лише числовими відліками у дискретні моменти часу .Процес перетворення неперервного сигналу у послідовність чисел називається дискретизацією.

Якщо частота дискретизації дорівнює f_s (кількість вимірів за секунду), то період відліку (час між двома вимірюваннями) визначається як:

$$T_s = 1 / f_s$$

Тоді дискретний сигнал можна описати формулою:

$s[n] = s(nT_s)$, де $n = 0, 1, 2, \ldots, N-1$

де s(t) — неперервний сигнал, s[n] — його дискретні відліки, n — кількість відліків у сигналі. Наприклад, якщо f_s = 16000 Гц, то за одну секунду ми отримаємо 16 000 відліків.

2. Приклад у вигляді масиву

Розглянемо простий приклад, який ілюструє, як неперервний звуковий сигнал перетворюється у цифровий вигляд.

Припустимо, ми беремо короткий фрагмент — 1 мс (0.001 секунди) звукової хвилі, і частота дискретизації становить 8 кГц, тобто f_s = 8000 Гц. Це означає, що за кожну секунду система робить 8000 вимірювань амплітуди сигналу.

Якщо взяти інтервал у 1 мс, тобто 1/1000 секунди, то за цей час отримаємо:

$$8000 \times 0.001 = 8 \text{ відліків.}$$

Іншими словами, неперервна хвиля розбивається на вісім окремих точок — моментів часу, у яких фіксується значення амплітуди. Так формується дискретна послідовність:

$$s = [s(0), s(1), s(2), \ldots, s(7)]$$

Для прикладу, якщо вхідний сигнал — це синусоїда з певною частотою, то послідовність відліків може виглядати так:

$$[0.00, 0.59, 0.95, 0.95, 0.59, 0.00, -0.59, -0.95]$$

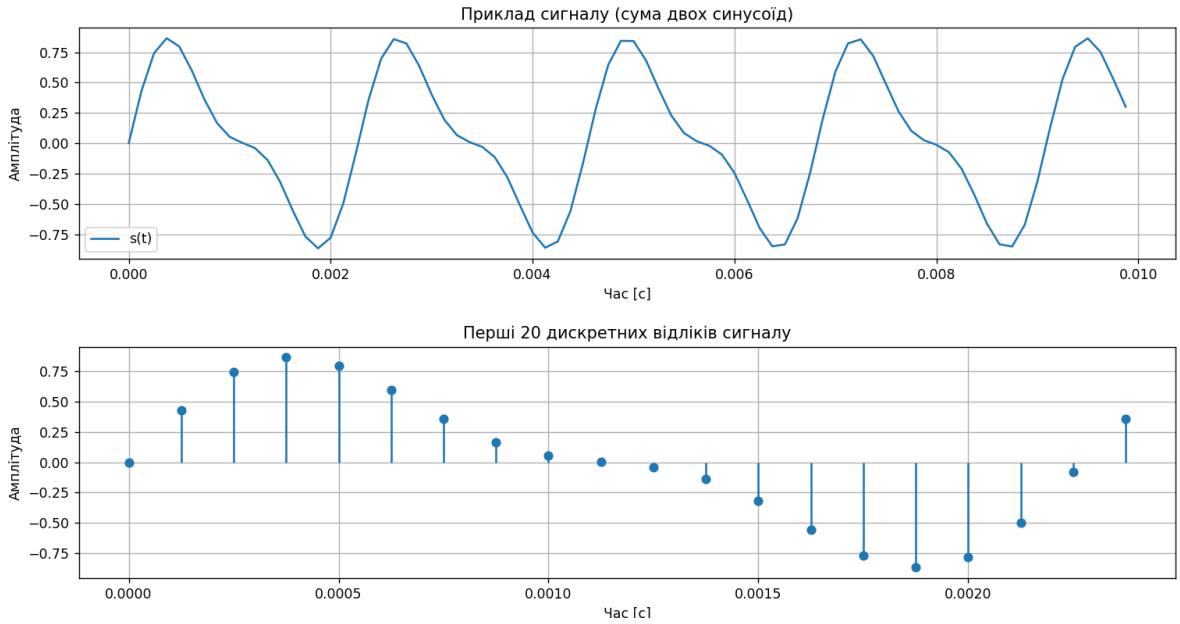
Кожен елемент цього масиву — це амплітуда звуку у відповідний момент часу. Завдяки такому поданню сигналу, будь-яка звукова хвиля може бути описана у вигляді масиву чисел, придатного для математичної обробки або зберігання у пам’яті комп’ютера.

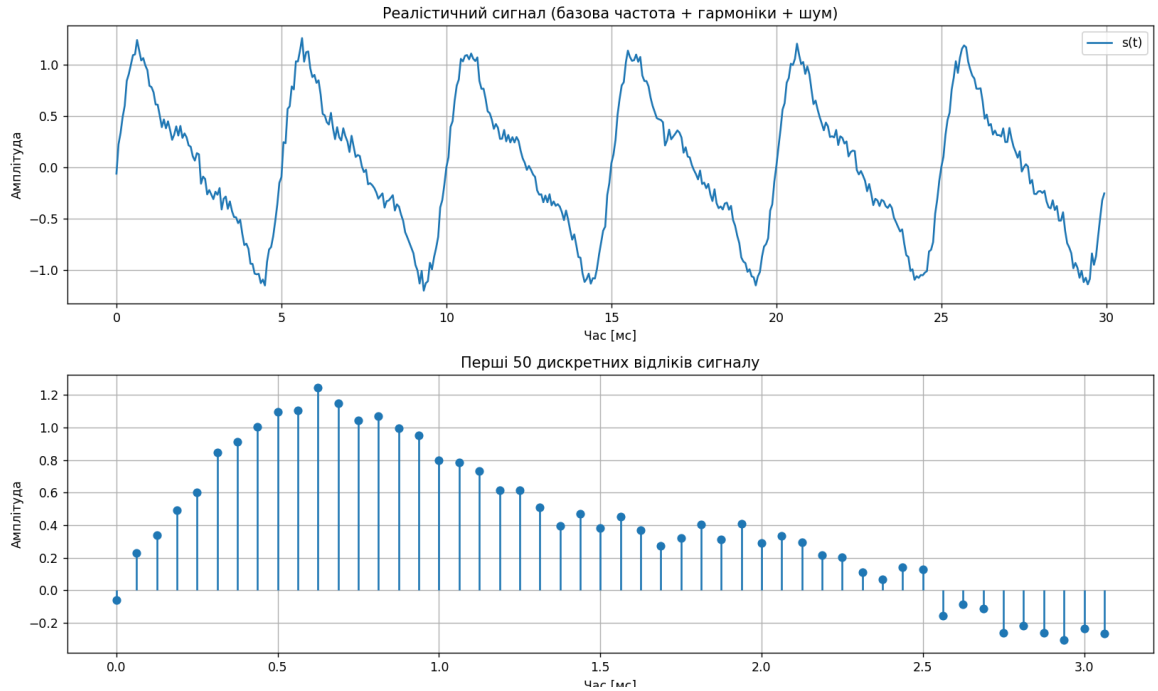
З математичної точки зору, такий масив є вектором дискретних значень, який можна розглядати як окремий набір спостережень неперервного процесу. З практичної точки зору, це — основа цифрового аудіо. Саме у

такому вигляді дані передаються у звукових кодеках, записуються на носії або подаються на вхід нейронних мереж.

Якщо відновити сигнал на графіку, точки масиву будуть відображені як зразки (samples), розташовані через однакові інтервали часу. При збільшенні частоти дискретизації (f_s) кількість таких точок на одиницю часу зростає, і цифрове представлення стає точнішим, тобто ближчим до реальної неперервної хвилі. Таким чином, масив s[n] є базовою формою представлення звукових даних у цифровій системі, де кожне число описує інтенсивність коливань у певний момент часу. Саме з цього етапу починається вся подальша обробка — фільтрація, спектральний аналіз, нормалізація або перетворення Фур'є.

Для наглядності можна переглянути два приклади де порівнюють вхідний сигнал з дискретизованим:





3. Теорема Найквіста–Шеннона

Одним із фундаментальних принципів цифрової обробки сигналів є теорема Найквіста–Шеннона. Вона визначає мінімальні умови, за яких процес дискретизації може точно відтворити початковий неперервний сигнал без втрати інформації.

Суть теореми полягає в тому, що будь-який неперервний сигнал можна відновити з його дискретних відліків, якщо частота дискретизації f_s є принаймні удвічі більшою за максимальну частоту f_{max}, присутню в сигналі:

$$f_s \geq 2 f_{max}$$

Ця нерівність отримала назву умова Найквіста і задає мінімально допустиму частоту вибірки для точного цифрового представлення сигналу. Якщо ж частота вибірки менша за цей поріг, виникає явище, відоме як аліасинг (aliasing) — коли високочастотні компоненти сигналу «накладаються» на низькочастотні, спотворюючи відновлений звук. У результаті система сприймає різні гармоніки як одну й ту саму частоту, і це призводить до помітного спотворення спектра.

Щоб краще зрозуміти цю закономірність, уявімо приклад. Припустимо, звуковий сигнал містить компоненти з частотами до 8 кГц. Згідно з теоремою Найквіста, мінімальна допустима частота дискретизації становить:

$$f_s = 2 \times 8\ кГц = 16\ кГц$$

Отже, щоб повністю зберегти інформацію про цей сигнал, потрібно виконувати щонайменше 16 тисяч вимірювань на секунду. Якщо вибрати меншу частоту, наприклад 10 кГц, частина спектра (від 5 до 8 кГц) «згорнеться» у нижчу частину діапазону, і звук набуде спотворень — може з’явитися характерне деренчання або хибні тони.

У реальних системах цифрового запису, як-от мікрофони, аудіоінтерфейси чи мікроконтролери, завжди використовують частоту дискретизації з запасом вище подвоєної граничної частоти. Це дозволяє компенсувати похибки аналогових фільтрів і забезпечує стабільну якість сигналу. Наприклад: для мови зазвичай обирають f_s = 16 кГц, бо спектр людського голосу обмежений приблизно 7–8 кГц; для музики або високоякісного звуку використовують f_s = 44,1 кГц (як у форматі CD); для професійних аудіосистем — навіть 48 кГц або 96 кГц, що забезпечує дуже точну передачу всіх гармонік. Теорема Найквіста–Шеннона є одним із ключових законів цифрової сигналізації, адже саме вона визначає межу, за якою аналоговий звук переходить у цифрову форму без втрат. Теорема також слугує базою для розрахунку параметрів мікрофонів, аудіокодеків, систем збирання даних та навіть алгоритмів нейронних мереж, що працюють із голосом чи акустичними подіями.

4. Нормалізація даних

Після етапу дискретизації звуковий сигнал представлений у вигляді послідовності числових значень, які відображають миттєву амплітуду коливань. Однак отримані дані можуть мати дуже різний масштаб — залежно від обладнання, розрядності аналого-цифрового перетворювача (АЦП) і способу збереження. Щоб зробити такі дані зручними для подальшої обробки, виконують нормалізацію — тобто масштабування значень сигналу до певного стандартного діапазону.

У цифрових аудіосистемах (наприклад, формат WAV або PCM) амплітуди зазвичай подаються як цілі числа в межах від −32768 до +32767, що відповідає 16-бітному представленню. У цьому випадку 0 означає відсутність відхилення (спокій), позитивні значення — підвищення тиску (або напруги), а негативні — його зменшення. Таке представлення зручно для апаратних звукових пристроїв і систем зберігання, але не завжди підходить для подальших математичних операцій.

У наукових розрахунках і при обробці сигналів у нейронних мережах зручніше оперувати дійсними числами (типу float), нормалізованими до інтервалу [−1, 1]. Це робиться за допомогою простого масштабування:

$$x_{norm} = x / x_{max}$$

де x — початкове значення амплітуди, x_{max} — найбільше можливе значення сигналу (наприклад, 32767), x_{norm} — нормалізоване значення.

У результаті всі відліки сигналу перетворюються так, щоб найсильніше відхилення мало амплітуду 1 або −1, а всі інші значення пропорційно зменшуються. Така форма зручна для обчислень, бо дозволяє уникнути переповнення змінних і зберігає відносні пропорції сигналу.

Нормалізація має особливе значення для алгоритмів машинного навчання та цифрової обробки. Більшість математичних моделей — особливо нейронні мережі — вимагають вхідних даних, що мають однаковий масштаб. Якщо подати сирий ненормалізований сигнал із великими відхиленнями амплітуд, алгоритм може працювати нестабільно: ваги мережі оновлюватимуться нерівномірно, а градієнти зникати. Тому нормалізація є обов’язковим етапом перед будь-яким навчанням моделей, класифікацією або спектральним аналізом.

Крім того, у системах реального часу нормалізація допомагає вирівняти рівень сигналу, зменшити вплив шумів і запобігти перевантаженню при подальшій обробці. Таким чином, цей процес виконує подвійну роль — забезпечує математичну стабільність алгоритмів і технічну узгодженість даних у різних середовищах.

Розклад у ряд Фур’є

1. Ряд Фур’є

Будь-який періодичний сигнал, навіть якщо він має складну або нерегулярну форму, можна подати у вигляді суми (суперпозиції) простих гармонічних коливань — синусоїд і косинусоїд різної частоти, амплітуди та фази. Такий підхід називається розкладом у ряд Фур’є, і він лежить в основі сучасних методів обробки сигналів, включно із цифровим аналізом звуку, синтезом мови та спектральною фільтрацією.

Для звукової хвилі це означає, що будь-який складний звук можна уявити як суму простих синусоїд, кожна з яких має свою частоту, амплітуду та фазу. Це дозволяє перейти від часової області (де ми спостерігаємо зміну сигналу у часі) до частотної області (де сигнал описується через набір частотних компонент). Формально, якщо задано періодичну функцію f(t) з періодом T, її можна подати як:

$$f(t) = a_0 + \sum_{n=1}^{\infty} [\, a_n \cdot cos(2\pi n \cdot t / T) + b_n \cdot sin(2\pi n \cdot t / T) \,]$$

де коефіцієнти ряду Фур’є визначаються так:

$$a_n = (2 / T) \int_0^T f(t) \cdot cos(2\pi n \cdot t / T) \cdot dt$$

$$b_n = (2 / T) \int_0^T f(t) \cdot sin(2\pi n \cdot t / T) \cdot dt$$

а середнє (постійне) значення сигналу обчислюється за формулою:

$$a_0 = (1 / T) \int_0^T f(t) \cdot dt$$

Тут T — період сигналу, а основна частота визначається як $f_0 = 1 / T$.

Кожна гармоніка має частоту $f_n = n \cdot f_0$, тобто є кратною основній. Таким чином, ряд Фур'є дозволяє представити будь-яку складну періодичну хвилю як набір гармонік.

У цифровій обробці сигналів, коли звук подано у вигляді дискретних відліків, застосовують дискретне перетворення Фур'є (DFT), яке є основою алгоритму швидкого перетворення Фур'є (FFT).

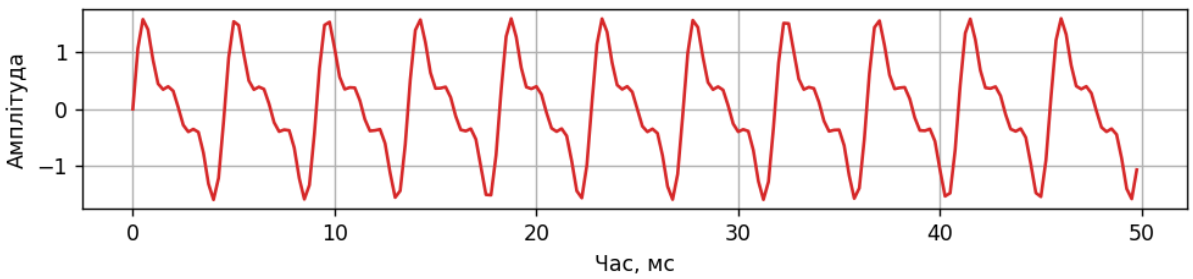
2 .Приклад розкладу у ряд Фур'є

У цьому прикладі синтезовано звуковий сигнал, що є сумою трьох гармонічних коливань із частотами $f_1 = 220$ Гц, $f_2 = 440$ Гц та $f_3 = 660$ Гц. Кожна з цих компонент є простою синусоїдою, а їхня сума формує більш складну хвилю, подібну до звучання акорду. Такий сигнал можна математично описати рівнянням:

$$f(t) = \sin(2\pi \cdot 220 \cdot t) + 0.6 \cdot \sin(2\pi \cdot 440 \cdot t) + 0.4 \cdot \sin(2\pi \cdot 660 \cdot t)$$

де коефіцієнти (1.0, 0.6, 0.4) задають амплітуди окремих гармонік.

Для цифрового подання було вибрано частоту дискретизації $f_s = 4000$ Гц, що відповідає періоду дискретизації $T_s = 1 / f_s = 0.00025$ с. Ця частота є достатньо високою, щоб задовольнити умову Найквіста ($f_s \geq 2 \cdot f_{max}$), тому сигнал не зазнає спотворення.



На часовому графіку видно періодичну структуру складного сигналу. Його форма суттєво відрізняється від звичайної синусоїди, оскільки одночасна дія кількох гармонік призводить до зміни амплітуди в часі. Саме так утворюється багатий «тембр» звуку.

Розклад у ряд Фур'є дозволяє показати, що цей складний сигнал є суперпозицією простих синусоїдальних складових. Для кожної гармоніки можна обчислити власні коефіцієнти:

$$a_n = (2 / T) \int_0^T f(t) \cdot cos(2\pi n \cdot t / T) \cdot dt$$

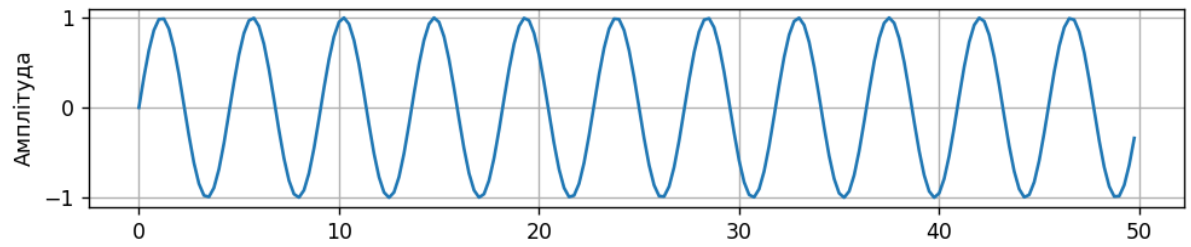
$$b_n = (2 / T) \int_0^T f(t) \cdot sin(2\pi n \cdot t / T) \cdot dt$$

а сам сигнал представити у вигляді:

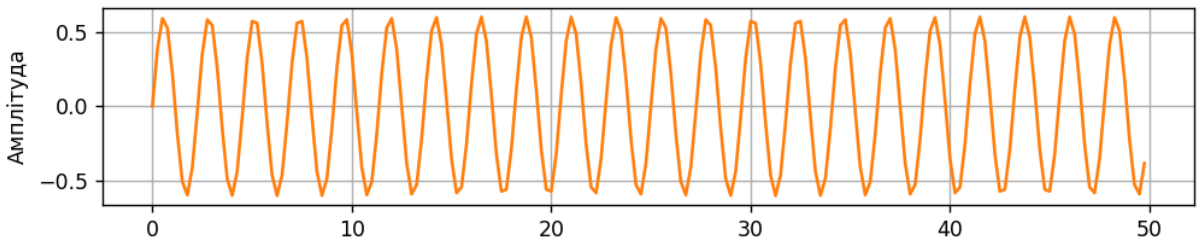
$$f(t) = a_0 + \sum_{n=1}^{\infty} [\, a_n \cdot cos(2\pi n \cdot t / T) + b_n \cdot sin(2\pi n \cdot t / T) \,]$$

де T — період сигналу, а n — номер гармоніки.

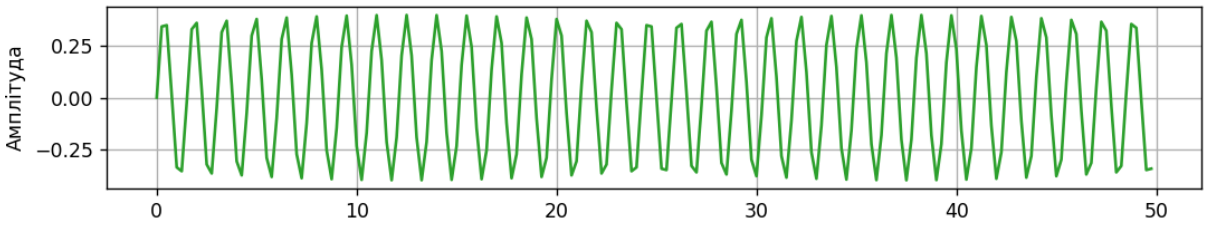
У нашому випадку, якщо виконати аналітичний або чисельний розклад, отримаємо три домінантні гармоніки, що відповідають частотам 220, 440 і 660 Гц. Тобто, складна хвиля фактично складається лише з трьох синусоїд.



220 Гц



440 Гц



660 Гц

Цей приклад демонструє головну ідею розкладу у ряд Фур’є: будь-який періодичний сигнал можна подати як суму простих гармонічних коливань. Завдяки цьому підходу можна переходити від часової форми сигналу до його частотної структури — визначати, які саме частоти беруть участь у формуванні звуку та з якою амплітудою.

Отже, розклад у ряд Фур’є — це математична основа, що дозволяє перетворювати хаотичну звукову хвилю у впорядкований набір частотних ознак. Саме завдяки йому можна аналізувати, розпізнавати та синтезувати звук, а в контексті нейронних мереж — навчати моделі «чути» та розуміти звукові сигнали.

3. Дискретне перетворення Фур'є (DFT)

Дискретне перетворення Фур'є (DFT, англ. Discrete Fourier Transform) є фундаментальним методом аналізу цифрових сигналів, який дозволяє представити часову послідовність вибірок у вигляді суми гармонічних компонент різної частоти. На відміну від класичного розкладання Фур'є для неперервних функцій, DFT застосовується до дискретних та скінченних наборів даних – саме таких, які формуються у цифрових системах збирання та обробки сигналів. Нехай маємо дискретний сигнал, представлений скінченною послідовністю:

$$x_0, x_1, x_2, \ldots, x_{(n-1)}$$

Тоді пряме дискретне перетворення Фур'є визначається формулою:

$$X_n = \sum_{k=1}^{N-1} x_k \cdot e\text{^}(-j \cdot 2\pi \cdot n \cdot k \,/\, N)$$

де: N — довжина сигналу, n — номер частотної компоненти, e^(−j·2π·n·k / N) — комплексна експонента, яка задає синусоїдальну основу.

Інверсне (зворотне) перетворення, що дозволяє відновити початковий сигнал з його спектра:

$$x_k = (1 \,/\, N) \sum_{n=1}^{N-1} X_n \cdot e\text{^}(+j \cdot 2\pi \cdot n \cdot k \,/\, N)$$

Ці дві формули є фундаментальними, вони встановлюють відповідність між часовою та частотною областями. DFT аналізує сигнал як суму базових комплексних синусоїд різних частот. Кожен коефіцієнт X_n говорить наскільки сильно в сигналі присутня частота, що відповідає індексу n. Іншими словами, DFT розкладає сигнал на спектр, перетворюючи часову

інформацію на частотну. Оскільки сигнал дискретний і скінченний, базові синусоїди також є дискретними та мають довжину N, тому утворюється замкнена ортогональна система.

Дискретне перетворення Фур'є, подане у вигляді суми. Для кожного значення n потрібно виконати N множень і N додавань. Але таких значень n також N. Тому загальна кількість операцій визначається як $O(N^2)$ Тобто алгоритм має квадратичну складність.

4. Приклад обчислення DFT для простого сигналу

Візьмемо короткий сигнал з довжиною N = 4: $x_0 = 1 x_1 = 0$ $x_2 = -1$ $x_3 = 0$ Це дуже проста послідовність, але вона показує суть гармонік у реальних DFT. Формула DFT для кожного n = 0, 1, 2, 3:

$$X_n = \sum_{k=1}^{N-1} x_k \cdot e\text{^}(-j \cdot 2\pi \cdot n \cdot k / N)$$

У нашому випадку:

$$X_n = \sum_{k=0}^{3} x_k \cdot e\text{^}(-j \cdot 2\pi \cdot n \cdot k / 4)$$

Обчислюємо X_0, підставляємо n = 0:

$$X_0 = \sum_{k=0}^{3} x_k \cdot e\text{^}(-j \cdot 2\pi \cdot 0 \cdot k / 4)$$

Оскільки $e^0 = 1$

$$X_0 = x_0 + x_1 + x_2 + x_3$$

$$X_0 = 1 + 0 + (-1) + 0 = 0$$

Обчислюємо X_1, підставляємо n = 1:

$$X_1 = \sum_{k=0}^{3} x_k \cdot e\text{^}(-j \cdot 2\pi \cdot 1 \cdot k / 4)$$

$$X_1 = 1 + 0 + 1 + 0 = 2$$

Пройшовши по усім значенням n маємо підсумковий спектр сигналу Маємо:

$$X_0 = 0$$

$$X_1 = 2$$

$$X_2 = 0$$

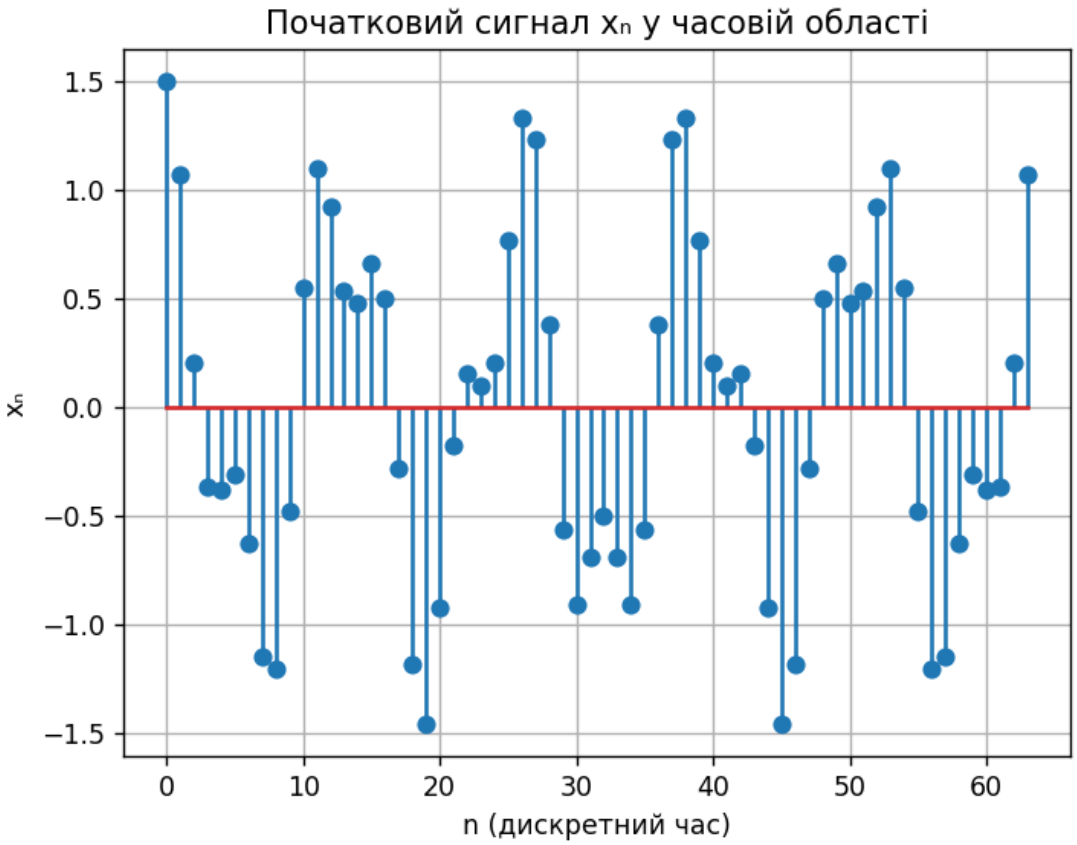
$$X_3 = 2$$

Висновок − Сигнал: 1, 0, −1, 0 має дві гармоніки на частоті n = 1, n = 3. Обидві з амплітудою 2. Це означає, що сигнал — це «чиста» синусоїда середньої частоти (половина частоти дискретизації).

5. Графічне представлення DFT

Метою даного експерименту є демонстрація процесу обчислення дискретного перетворення Фур'є (DFT) для штучно сформованого сигналу, що складається з двох гармонічних компонент різної частоти.

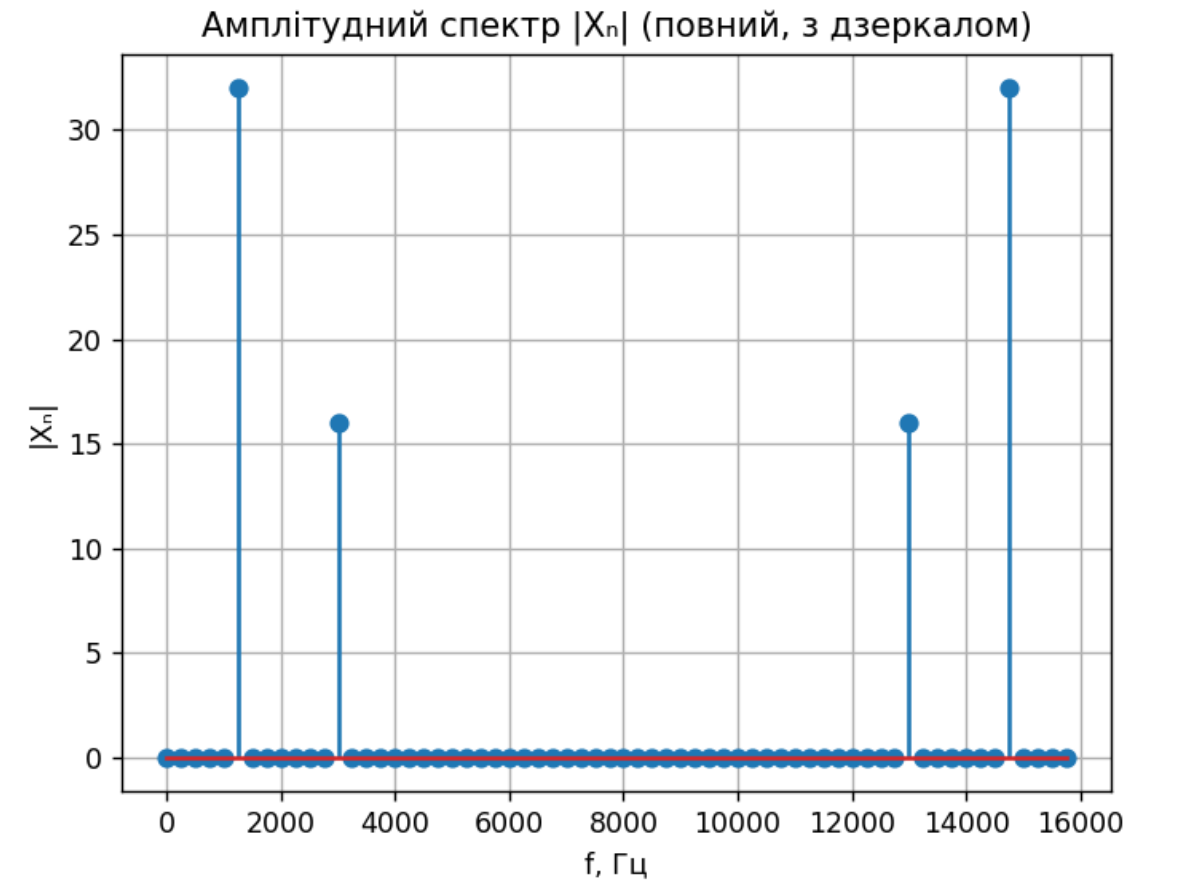
Це дозволяє наочно дослідити механізм переходу від часової області до частотної, а також підтвердити, що DFT коректно відображає частотний склад сигналу. Для експерименту було узято дискретний сигнал довжиною $N = 64$, частотою дискретизації $f_s = 16\,000$ Гц. Сигнал складається із суми двох косинусних компонент. Перша гармоніка з частотою $f_1 = 1250$ Гц і амплітудою 1.0 Друга гармоніка з частотою $f_2 = 3000$ Гц і амплітудою 0.5**.** Це дозволяє отримати спектр з двома добре видимими піками.



Після формування сигналу було обчислено його дискретне перетворення Фур'є через реалізацію DFT.

$$X_n = \sum_{k=0}^{63} x_k \cdot e\text{^}(-j \cdot 2\pi \cdot n \cdot k \,/\, 64)$$

Результати були візуалізовані у вигляді графіка.



Результат DFT

Після дискретного перетворення Фур'є ми отримуємо багато значень:

$$X_0, X_1, X_2, \ldots, X_{(n-1)}$$

Це комплексні числа. Тобто кожне X_n має вигляд:

$$X_n = Re + j \cdot Im$$

де Re — дійсна частина, Im — уявна частина. У більшості практичних застосувань, особливо під час аналізу аудіосигналів, використовується модуль комплексного коефіцієнта X_n

$$|X_n| = \sqrt{(Re(X_n)^2 + Im(X_n)^2)}$$

З фізичного погляду величина $|X_n|$ відповідає амплітуді гармонічної складової сигналу з частотою f_n, де:

$$f_n = n \cdot f_s / N$$

STFT / FFT

1. Швидке перетворення Фур'є

Метод швидкого перетворення Фур'є (Fast Fourier Transform, або FFT) є фундаментальним інструментом цифрової обробки сигналів. Він дозволяє швидко і ефективно перейти від часового опису сигналу (де відображається зміна його амплітуди в часі) до частотного опису, який показує, з яких гармонічних складових складається сигнал і з якою інтенсивністю вони присутні. Історично, класичне дискретне перетворення Фур'є (DFT) вимагало виконання N^2 математичних операцій для сигналу з N відліків, що було надзвичайно повільним для реальних застосувань. Алгоритм FFT, запропонований Коулі та Тьюкі у 1965 році, дозволив зменшити складність до $N \cdot \log_2 N$, тобто зробив спектральний аналіз практично миттєвим навіть для тисяч відліків. Завдяки цьому FFT став основою всіх сучасних методів обробки звуку, зображень, радіосигналів та вібраційних даних. Ідея полягає в тому, що сигнал можна розділити на парні й непарні елементи, обчислити дві менші суми, а потім їх об'єднати. Запишемо розбиття DFT на дві частини:

$$X_n = \sum_{k=0}^{N/2-1} x_{2k} \cdot e\text{^}(-j \cdot 2\pi \cdot n \cdot 2k \;/\; N) + e\text{^}(-j \cdot 2\pi \cdot n \;/\; N) \cdot \sum_{k=0}^{N/2-1} x_{2k+1} \cdot e\text{^}(-j \cdot 2\pi \cdot n \cdot 2k \;/\; N)$$

Тут перша сума відповідає парним відлікам x_{2k}, а друга - непарним x_{2k+1}. Для зручності вводять позначення:

$$E_n = \sum_{k=0}^{N/2-1} x_{2k} \cdot e\text{^}(-j \cdot 2\pi \cdot n \cdot k \;/\; (N/2))$$

$$O_n = \sum_{k=0}^{N/2-1} x_{2k+1} \cdot e\text{^}(-j \cdot 2\pi \cdot n \cdot k \;/\; (N/2))$$

Тоді загальне рівняння набуває компактного вигляду:

$$X_n = E_n + W_n \cdot O_n$$

де $W_n = e^{(-j \cdot 2\pi \cdot n / N)}$ — обертальний множник. Використовуючи властивість періодичності експоненти, можна отримати значення другої половини спектра без додаткових розрахунків:

$$X_{n+n/2} = E_n - W_n \cdot O_n$$

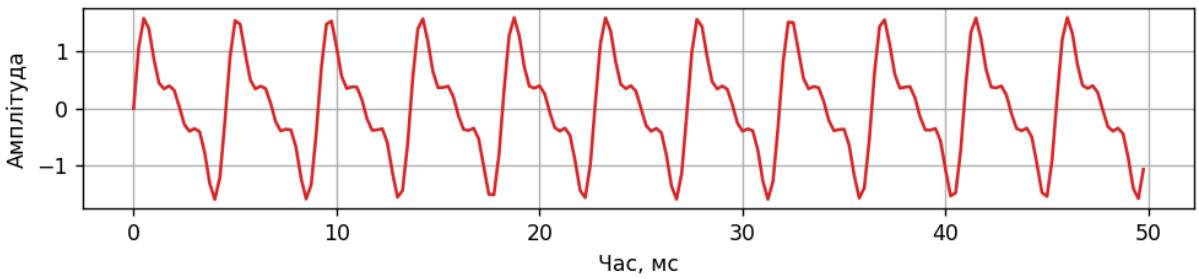
Тут важливо, що E_n і O_n самі по собі є DFT, але вже для послідовностей довжини N/2, і їх можна обчислювати за тим самим алгоритмом рекурсивно. Таким чином, одна велика задача розкладається на дві задачі удвічі меншої розмірності, потім кожна з них ще на дві, і так далі, поки не дійдемо до елементарних випадків довжини 2 або 4, де перетворення можна обчислити буквально кількома операціями. FFT - це не нова формула, а спосіб швидше обчислити ту саму суму, використовуючи властивості симетрії та періодичності експоненти $e^{(-j \cdot 2\pi \cdot n \cdot k / N)}$. Завдяки такому розбиттю FFT виконує лише $N \cdot \log_2 N$ операцій замість N^2. Наприклад при N = 1024 звичайний DFT потребує понад 1 048 576 операцій. FFT — лише близько 10 240 це в 102 рази швидше

2. Приклад FFT

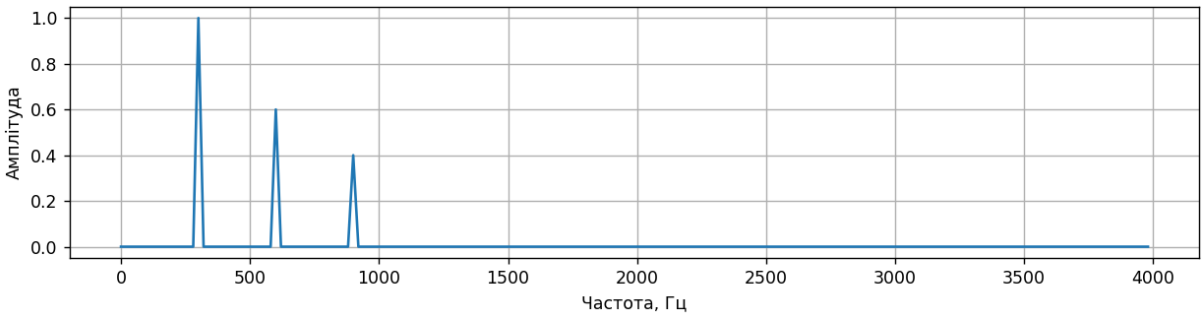
Для демонстрації принципу роботи швидкого перетворення Фур'є візьмемо раніше використаний сигнал, який описується як сума трьох синусоїдальних хвиль різної частоти:

$$x(t) = \sin(2\pi \cdot 300 \cdot t) + 0.6 \cdot \sin(2\pi \cdot 600 \cdot t) + 0.4 \cdot \sin(2\pi \cdot 900 \cdot t)$$

Перший доданок має частоту 300 Гц і є основною гармонікою, другий (600 Гц) та третій (900 Гц) — вищими гармоніками з меншими амплітудами 0.6 та 0.4 відповідно. У результаті формується складна періодична хвиля, подібна до тону із трьома домінантними частотами.



Алгоритм FFT перетворює часовий сигнал у частотну область, обчислюючи, які гармонічні компоненти (частоти) входять до його складу та з якою інтенсивністю. Отриманий спектр показує, які частоти є в сигналі та з якою силою вони виражені. Добре видно три виразні піки на частотах 300 Гц, 600 Гц та 900 Гц, що відповідають гармонікам, з яких складається сигнал.

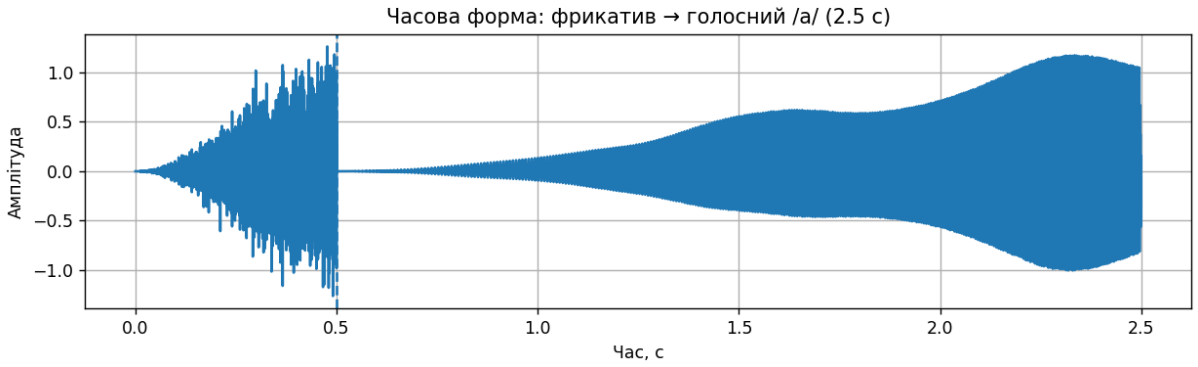


Висота кожного піка пропорційна амплітуді відповідної гармоніки - найбільший пік при 300 Гц, менший при 600 Гц і ще менший при 900 Гц. Таким чином, цей приклад демонструє, як FFT дозволяє “розкласти” складний сигнал на його частотні складові, виявляючи основну структуру звукової хвилі.

Метод швидкого перетворення Фур’є дає можливість швидко виконати спектральний аналіз навіть у реальному часі, що є ключовим етапом у системах цифрової обробки звуку, розпізнавання мови та нейромережевих алгоритмах аудіоаналізу.

Для більш наочного прикладу було змодельовано складний акустичний сигнал, який за структурою наближений до природного мовного звука.

Такий сигнал складається з двох послідовних фрагментів: фрикативного шуму (приблизно 0.5 с) та озвученого голосного /a/ (приблизно 2 с), у якого поступово підвищується основна частота f_0 від 110 до 190 Гц. Таким чином, формується модель звука, подібного до переходу від приголосного “с” до голосного “а”.



3. Короткочасне перетворення Фур’є

Однак звичайне перетворення Фур’є (і його дискретна форма) має один суттєвий недолік: воно показує, які частоти є присутні у сигналі, але не коли вони з’являються. У випадку звукових хвиль це означає, що ми знаємо спектральний склад, але не бачимо динаміки — як змінюються частоти в часі. Для мовних сигналів це критично, адже звуки мови (фонеми, інтонації, наголоси) короткочасні й постійно змінюються.

Щоб подолати цю обмеженість, було розроблено метод короткочасного перетворення Фур’є (STFT). Його ідея полягає в тому, що весь сигнал ділиться на короткі відрізки — вікна (зазвичай тривалістю від 10 до 50 мс). Для кожного такого вікна обчислюється своє перетворення Фур’є, а отримані спектри розміщуються послідовно у часі.

Це дає можливість аналізувати, як частотний склад сигналу змінюється у часі. Наприклад, у спектрограмі людського голосу можна чітко побачити появу та зникнення формант, гармонік або шумових ділянок. Завдяки цьому STFT є основним інструментом у системах розпізнавання мовлення, музичному аналізі, фільтрації шумів та навіть у дослідженнях біоакустики. STFT сигнал x(t) визначається як:

$$S(t, \omega) = \int_{-\infty}^{+\infty} x(\tau) \cdot w(\tau - t) \cdot e\text{^}(-j \cdot \omega \cdot \tau) \cdot d\tau$$

де x(τ) — аналізований сигнал; w(τ − t) — віконна функція, центрована в моменті часу t; e^(−j·ω·τ) — комплексна експонента, яка визначає перетворення Фур’є.

У цифровому вигляді ця формула переходить у дискретну форму:

$$X_k[i] = \sum_{k=0}^{N/2-1} x_k \cdot w_{(k} - t_{0)} \cdot e\text{^}(-j \cdot 2\pi \cdot f_n \cdot k \,/\, f_s)$$

x_k — вхідний дискретний сигнал $w_{(k}-t_{0)}$ — віконна функція (Hann, Hamming, Blackman…) t_0 — часова позиція центра або початку вікна. $f_n = n \cdot f_s/N$). N — довжина вікна. f_s — частота дискретизації. e^(−j·2π·f_n·k/F_s) — комплексна експонента. Оскільки STFT виконується на обмежених ділянках сигналу, важливо забезпечити плавність переходу між кадрами, щоб уникнути спотворень спектра. Для цього вводиться віконна функція, яка згладжує краї кожного фрагмента.

Після виконання короткочасного перетворення Фур'є ми отримуємо комплексну матрицю коефіцієнтів S(t, ω), де кожен елемент має дійсну частину Re(S) і уявну частину Im(S).

$$S(t, \omega) = Re(S) + j \cdot Im(S)$$

Такий запис описує не лише "силу" (амплітуду) певної частоти, а й фазовий зсув, який визначає, у який момент часу ця гармоніка має максимум. Щоб отримати реальну величину, яка відображає "силу" або "амплітуду" гармоніки, обчислюється модуль цього комплексного числа:

$$|S(t, \omega)| = \sqrt{(Re(S(t, \omega))^2 + Im(S(t, \omega))^2)}$$

Це те саме, що знайти довжину вектора у комплексній площині, який має координати (Re, Im). Таким чином, |S(t, ω)| є дійсною величиною, що відображає енергію або амплітуду сигналу на частоті ω в момент часу t. Сирі амплітудні значення часто мають дуже широкий діапазон: одні частоти можуть мати амплітуду 1, інші — 0.0001. У такому вигляді спектрограма виглядає непридатною для візуалізації — потужні частоти "перекривають" слабкі, і дрібні деталі просто губляться. Щоб це виправити, застосовується логарифмічне перетворення — аналогічне тому, як працює шкала децибелів у фізиці звуку. Логарифм "стискає" великий діапазон значень, роблячи слабкі частоти помітними.

Логарифмічна спектрограма визначається як:

$$S\_dB(t, \omega) = 20 \cdot \log_{10}(|S(t, \omega)|)$$

Число 20 тут походить із фізичного означення рівня звукового тиску: рівень у децибелах визначається за формулою $20 \cdot \log_{10}(A_1 / A_0)$, де A_1 — поточна амплітуда, A_0 — базовий рівень (референс). Таким чином, збільшення амплітуди у 10 разів відповідає зростанню на 20 дБ, а зменшення у 10 разів — падінню на 20 дБ.

Припустімо, що для певної частоти ω_1 на момент часу t_1 $|S(t_1, \omega_1)| = 0.1$, а для іншої частоти ω_2 $|S(t_1, \omega_2)| = 0.001$. Якщо взяти ці значення напряму, різниця у 100 разів здається надто великою. Але після переходу до децибелів:

$$S\_dB(t_1, \omega_1) = 20 \cdot \log_{10}(0.1) = -20 \text{ дБ}$$
$$S\_dB(t_1, \omega_2) = 20 \cdot \log_{10}(0.001) = -60 \text{ дБ}$$

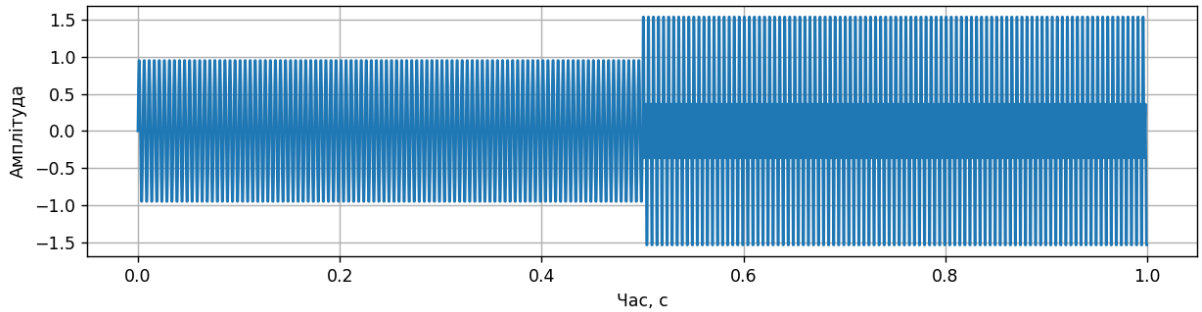
Різниця між ними тепер складає 40 дБ, що є зручним діапазоном для відображення на графіку. Тому слабкі частоти не зникають, а залишаються видимими на спектрограмі.

Таке зображення дає змогу одночасно спостерігати часові та спектральні характеристики сигналу. Саме тому спектрограма є основним інструментом аналізу складних, непостійних у часі звукових процесів — наприклад, елементів мовлення, музичних фрагментів, природних звуків або шумів технічного походження.

4. Приклад реалізації STFT

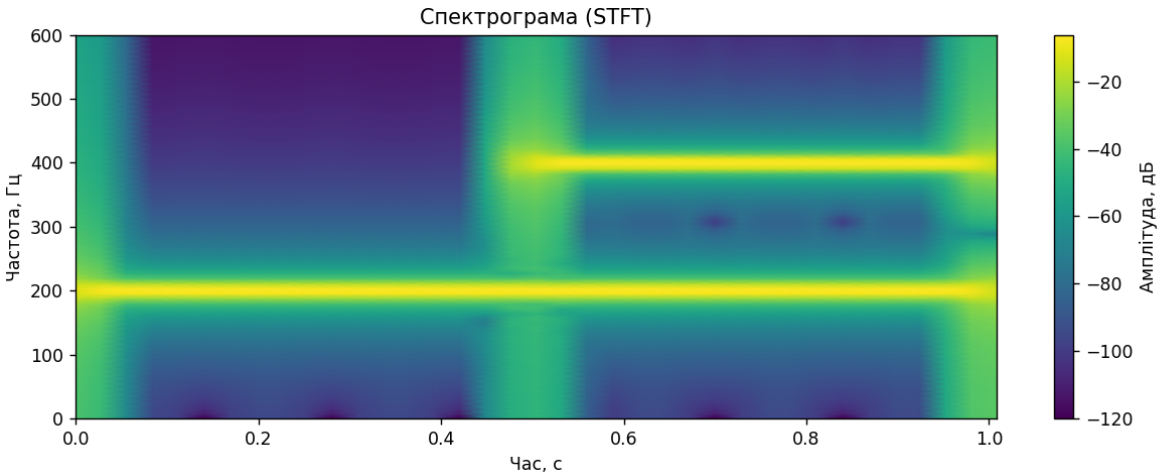
Для прикладу розглянемо простий сигнал, який містить дві гармонічні складові: $x(t) = \sin(2\pi \cdot 200 \cdot t) + \sin(2\pi \cdot 400 \cdot t) \cdot u(t - 0.5)$

де u(t) — одинична функція Хевісайда, яка «вмикає» другу складову після моменту часу t = 0.5 с. У перші пів секунди сигнал містить лише одну частоту 200 Гц, а після 0.5 с додається друга частота 400 Гц.(графік 1)



графік 1

Якщо виконати STFT і побудувати спектрограму, то на зображенні до t = 0.5 с буде видно одну горизонтальну смугу (відповідає 200 Гц), а після — дві (200 і 400 Гц). Таким чином, спектрограма дозволяє наочно побачити момент появи нової гармонічної компоненти та прослідкувати, як змінюється структура сигналу.



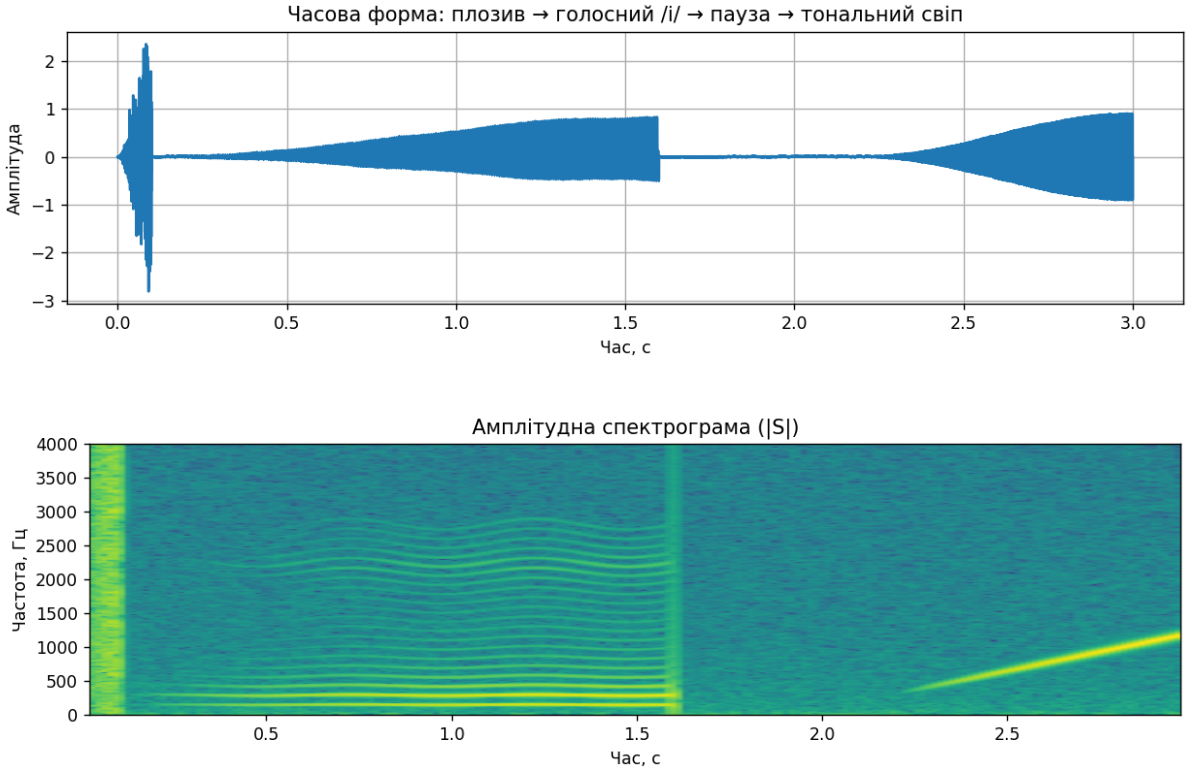
графік 2

до 0.5 с наявна лише одна частота 200 Гц, а після цього з’являється друга — 400 Гц. Таким чином, спектрограма відображає розвиток частот у часі, що неможливо побачити з простої часової форми сигналу.

5. Ускладнений приклад STFT (Реалістичний)

Для демонстрації короткочасного перетворення Фур’є (STFT) було змодельовано акустичний сигнал тривалістю 3 секунди при частоті дискретизації. $f_s = 16000$ Гц, де період дискретизації визначається як

$$T_s = 1 / f_s.$$



Сигнал складається з чотирьох послідовних епізодів, кожен з яких моделює певний тип звукового фрагмента, характерного для реального мовлення або техногенних акустичних подій.

Перший фрагмент сигналу (0–0.10 с) представляє собою короткий широкосмуговий імпульсний сплеск, який імітує вибуховий приголосний на зразок [p], [t] або [k]. На спектрограмі цей компонент проявляється як

коротка по часу, але широка по частоті пляма, що відображає одночасну присутність енергії у всьому частотному діапазоні. Такий тип сигналу має переважно шумову природу та різкий фронт амплітуди, характерний для моменту видиху повітря під час утворення приголосного звуку.

Другий епізод (0.10–1.60 с) моделює озвучений голосний звук /i/. Джерело сигналу - періодичний коливальний процес із фундаментальною частотою $f_0 \approx 140$ Гц, до якої додається невелике вібрато ±4 Гц, що імітує природні коливання висоти голосу людини.

На спектрограмі у цьому інтервалі спостерігається система горизонтальних гармонічних смуг, які відповідають кратним частотам f_0, при цьому інтенсивність сигналу зростає поблизу формант. Для уникнення різких переходів було додано амплітудну огинаючу типу “атака–спад”, що забезпечує плавне наростання та згасання звуку.

У третій частині сигналу (1.60–2.20 с) моделюється коротка пауза. Цей фрагмент містить лише мінімальний рівень фонового шуму, тому на спектрограмі він виглядає як зона низької енергії, практично позбавлена спектральних ліній. Така ділянка є типовою для природного мовлення, де між озвученими звуками існують короткі відрізки мовчання або дихання.

Заключний етап моделі (2.20–3.00 с) являє собою тональний “сиренний” свіп — синусоїдальний сигнал, частота якого лінійно зростає від 300 Гц до 1200 Гц упродовж останніх 0.8 секунди. На спектрограмі цей фрагмент виглядає як чітка похила лінія, що ілюструє поступове зміщення частоти з часом. Отримані результати підтверджують, що спектрограма є надзвичайно інформативним інструментом для візуального та кількісного аналізу звукових сигналів. На відміну від класичного перетворення Фур’є, яке показує лише загальний спектральний склад, STFT дозволяє бачити динаміку частотних компонент, тобто як енергія різних частот змінюється

у часі. Це має принципове значення для розпізнавання мовлення, класифікації аудіо подій, музичного аналізу, а також для побудови ознак, що використовуються у нейронних мережах (зокрема, при формуванні мел-спектрограм і MFCC).

Енергетичні та спектральні характеристики

Розрахунок енергетичних і спектральних характеристик є одним із ключових етапів аналізу акустичних сигналів. Якщо перетворення Фур'є та STFT дозволяють отримати розподіл енергії за частотами, то подальше обчислення інтегральних показників — таких як енергія, потужність, середньоквадратичне значення, спектральна щільність потужності (PSD) або відношення сигнал/шум (SNR) — дає змогу кількісно оцінити властивості сигналу. Такі параметри необхідні для порівняння, класифікації та контролю якості сигналів у різних прикладних задачах. Зокрема, вони дозволяють визначити, наскільки сильний, чистий або зашумлений сигнал, які частотні діапазони містять найбільшу енергію, а які — лише фонові перешкоди.

1. Попередня обробка та нормалізація сигналів.

Під час аналізу великої кількості аудіофайлів важливо мати однаковий енергетичний рівень. Для цього застосовують нормування за RMS або середньою потужністю, щоб уникнути перекосів у навчальних вибірках нейронних мереж або в статистичних моделях.

2. Оцінка якості запису.

Під час обробки звукових даних, особливо у задачах розпізнавання мови або класифікації аудіо, важливу роль відіграє якість запису. Основним кількісним показником цієї якості є співвідношення сигнал/шум — Signal-to-Noise Ratio (SNR). SNR характеризує, наскільки корисний сигнал переважає над фоновими перешкодами. Високе значення SNR означає, що запис чистий, з мінімальними спотвореннями та шумом. Низьке ж свідчить про домінування шуму, який може ускладнювати подальшу обробку, сегментацію або розпізнавання нейронною мережею. З фізичної точки зору, сигнал — це впорядкована частина енергії, яка несе інформацію

(наприклад, голос або музичний тон), тоді як шум — це випадкові флуктуації, які не несуть корисного змісту. SNR показує відношення енергії цих двох компонент. Для цифрового сигналу x[n], який складається з корисної частини s[n] та шуму n[n]:

$$x[n] = s[n] + n[n]$$

співвідношення сигнал/шум визначається як:

$$SNR = P_s / P_n = (\Sigma_n\, s[n]^2) / (\Sigma_n\, n[n]^2)$$

У логарифмічній шкалі, що зручна для практики, значення SNR виражається в децибелах:

$$SNR(dB) = 10 \cdot \log_{10}(P_s / P_n) = 20 \cdot \log_{10}(RMS_s / RMS_n)$$

де P_s — середня потужність корисного сигналу, P_n — середня потужність шуму, RMS_s, RMS_n — середньоквадратичні значення сигналу і шуму відповідно.

Практичні орієнтири для SNR

SNR > 40 дБ — дуже чистий запис (студійна якість, майже відсутній шум).
SNR ≈ 30 дБ — хороша якість, допустима для більшості систем розпізнавання мови.
SNR ≈ 20 дБ — задовільна якість, але шум може частково спотворювати спектр.
SNR < 10 дБ — значне зашумлення, потрібна попередня фільтрація або шумозниження.

У реальних акустичних умовах (побутові записи, мікрофони IoT, ESP32, мобільні сенсори) SNR зазвичай коливається від 10 до 35 дБ. В ідеальному випадку, якщо відома шумова складова (наприклад, під час калібрування

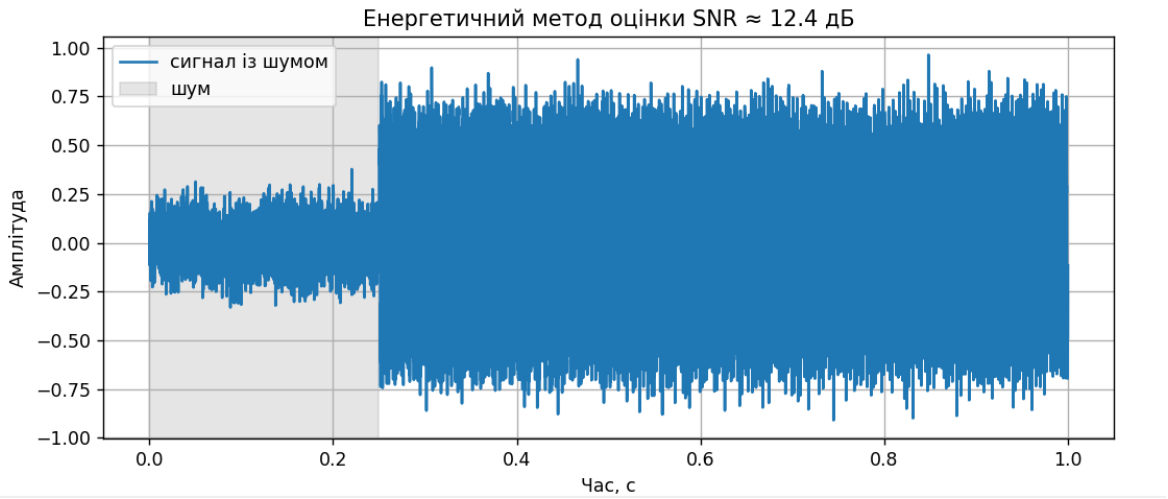
мікрофона), SNR можна обчислити безпосередньо. Але для реальних сигналів шум часто невідомий, тому використовують такі підходи:

Емпіричний (через ділянки тиші):

Вибирають інтервали, де сигнал відсутній (наприклад, між словами), і вважають їх шумом. Потім обчислюють SNR між енергією активної частини сигналу та цих «мовчазних» ділянок.

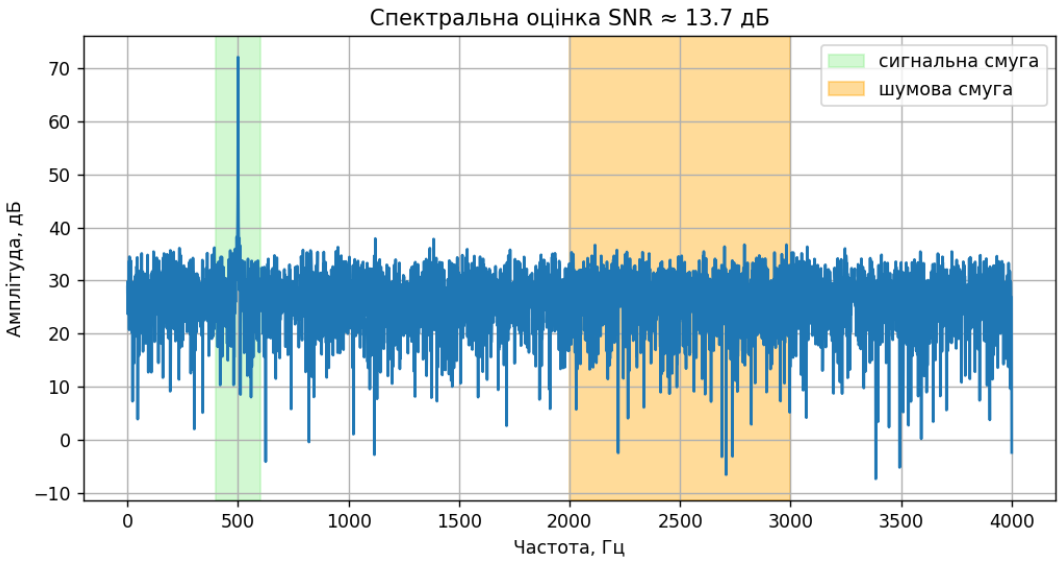
Приклад:

На графіку чітко видно першу ділянку — шумову, і другу — де сигнал з'являється. Розраховане значення SNR показує співвідношення між їхніми енергіями.



Модельний (через спектральне відношення):

Оцінюють спектр сигналу S(f) і спектр шуму N(f). Тоді: $SNR(f) = |S(f)|^2 / |N(f)|^2$. А інтегрування по всьому діапазону частот дає середнє значення SNR.



Методи на основі енергії фреймів:

Для кожного короткого вікна (наприклад, 25 мс) обчислюється енергія сигналу. Якщо енергія вища за поріг — це активний сигнал, нижча — шум. Далі середнє по класах дає оцінку SNR для запису. Безперервний сигнал x(t) або його дискретна форма x[n] ділиться на послідовні фрагменти довжиною 20–30 мілісекунд. У кожному такому фрагменті сигнал вважається практично незмінним за спектром та амплітудою. Для покращення розділення між фреймами застосовується вікно Ганна, яке згладжує переходи між межами фреймів і зменшує спектральні витоки.

Формула вікна Ганна має вигляд:

$$w[n] = 0.5 \cdot (1 - \cos(2\pi n / (L - 1)))$$

де L — кількість відліків у фреймі, n — індекс відліку ($0 \leq n < L$).

Для кожного фрейму сигналу обчислюється його енергія (середня потужність):

$$E_k = (1 / L) \cdot \Sigma (x_k[n] \cdot w[n])^2$$

де E_k — енергія k-го фрейму, $x_k[n]$ — відліки сигналу у k-му фреймі, $w[n]$ — значення вікна Ганна. Отримані значення енергії відображають інтенсивність сигналу у кожному часовому інтервалі. У випадку мовлення енергія голосових ділянок суттєво перевищує енергію шумових пауз. Для розділення фреймів на активні (сигнал) і неактивні (шум) використовується порогове значення енергії T. Найпростіший спосіб — обрати T як певну статистичну величину від розподілу енергій. Використано адаптивний підхід:

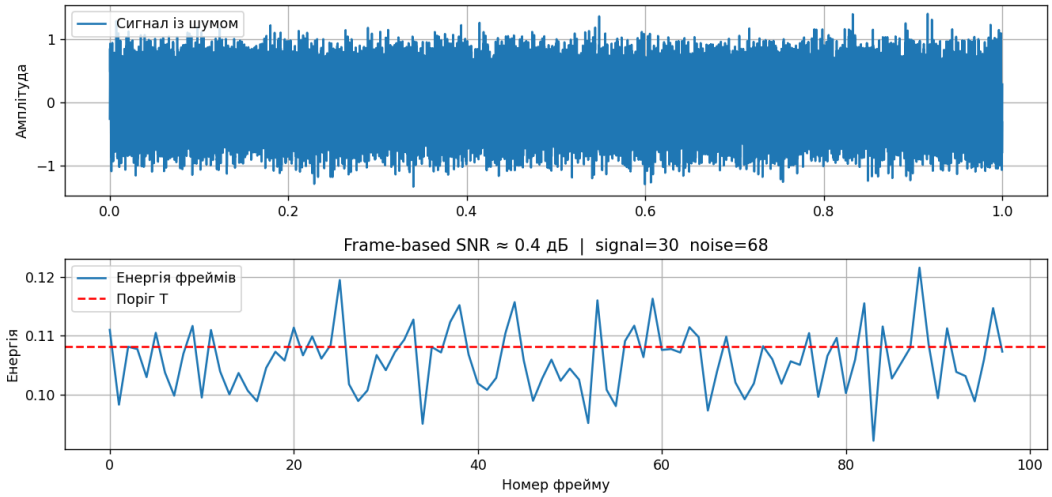
$$T = \mu + 0.5\sigma$$

де μ — середнє значення енергій усіх фреймів, σ — стандартне відхилення.

Фрейми, для яких $E_k > T$, вважаються сигналом, а ті, де $E_k \leq T$, — шумом.

Приклад:

Згенеровано тестовий сигнал — синусоїду частотою 440 Гц з доданим білим шумом. Сигнал дискретизовано з частотою 16 кГц, поділено на фрейми тривалістю 25 мс із кроком 10 мс. Для кожного фрейму розраховано енергію, а потім визначено поріг T і середні значення потужностей.Отримані графіки містять дві частини (граф. нижче). Верхній графік — часовий сигнал з шумом, який імітує запис голосу у шумному середовищі. Нижній графік — енергія кожного фрейму та поріг T (пунктирна червона лінія). Фрейми, що перевищують поріг, вважаються активними. За співвідношенням енергій між цими групами обчислюється SNR.



Метод оцінки SNR на основі енергії фреймів із застосуванням вікна Ганна є простим, але ефективним інструментом для кількісної оцінки якості звуку. Він забезпечує можливість виділення ділянок активності сигналу та визначення рівня шуму. Також даний метод використовується в алгоритмах Voice Activity Detection (VAD), у нейронних системах.

3. Виділення інформативних ділянок.

Будь-який аудіосигнал складається з чергування активних і пасивних ділянок. Активні ділянки характеризуються високою енергією, складною частотною структурою та швидкими змінами амплітуди. Пасивні — навпаки, мають низьку амплітуду та рівномірний спектр, типовий для фонових шумів або пауз. Задача полягає у виявленні моментів початку та завершення активності, що у мовних системах називається Voice Activity Detection (VAD). У системах розпізнавання мовлення, таких як Espressif AFE, модуль VAD автоматично визначає, коли користувач починає або закінчує говорити. Найпростіший і найпоширеніший спосіб полягає у використанні енергетичних характеристик фреймів. Сигнал x[n] ділиться на фрейми тривалістю L відліків, і для кожного фрейму обчислюється енергія (детальніше на ст. 33). На практиці енергетичний аналіз фреймів використовується як первинний детектор активності, після якого застосовуються більш складні методи — спектральні або нейронні.

4. Оцінка ефективності фільтрації.

Порівнюючи енергію сигналу до та після обробки (наприклад, після шумозниження чи еквалізації), можна визначити, наскільки корисна енергія збережена, а шум — знижений. Це дозволяє об’єктивно оцінювати роботу алгоритмів попередньої обробки.

5. Побудова ознак для машинного навчання.

Енергетичні й спектральні характеристики широко використовуються як фічі (features) для моделей класифікації звуків, розпізнавання мовлення або біоакустичного моніторингу. На основі енергії у вузьких смугах частот, спектрального центроїда, ширини чи «плоскісності» спектра можна створювати компактні числові вектори, які добре описують акустичний образ сигналу.

Перцептивні моделі (мел-шкала, MFCC)

Перцептивні моделі — це набір підходів і математичних перетворень, які намагаються відобразити те, як людина сприймає звук, а не лише те, як цей звук описується фізично (у герцах і паскалях). головна ідея проста: слух — нелінійний і вибірковий, тож якщо ми перетворимо аудіосигнал у «перцептивно узгоджені» ознаки, вони краще відповідатимуть реальній чутливості слуху й забезпечать більш точну, стійку роботу алгоритмів розпізнавання.

1. Мел-шкала

Наш слух — не лінійний прилад, і він не сприймає частоту прямо «в герцах». Якщо подати два звуки — один 200 Гц, інший 300 Гц — різниця буде помітна. Але якщо зробити те саме на високих частотах, наприклад 5200 і 5300 Гц, то вухо майже не відчує різниці. Це означає, що ми краще розрізняємо низькі частоти, а на високих — наша здатність до розрізнення погіршується.

Саме для цього й створили мел-шкалу (melody — «мелодія»): вона перетворює фізичну частоту у шкалу, ближчу до того, як наш мозок сприймає зміни висоти звуку. Іншими словами, вона «стискає» верхній діапазон і «розтягує» нижній, щоб відстань між двома точками на шкалі відповідала приблизно однаковій сприйманій різниці у висоті.

2. Формули для перетворення

Залежність між частотою f (у герцах) і мел-шкалою m описується формулою:

$$m = 2595 \cdot \log_{10}(1 + f / 700)$$

Зворотне перетворення (щоб з мела знову отримати герци):

$$f = 700 \cdot (10^{(m / 2595)} - 1)$$

Ці формули не просто математична зручність — вони наближено відтворюють експериментальні дані про те, як люди оцінюють висоту звуку при прослуховуванні синусоїд. Наприклад, збільшення частоти з 500 до 1000 Гц сприймається так само «помітно», як збільшення з 2000 до майже 4000 Гц. Тобто, у мел-шкалі перша половина (до ≈1000 Гц) описує дуже вузький діапазон звуків, який вухо сприймає тонко, а далі — значно ширше охоплення, де людська чутливість падає. Щоб «навчити» комп'ютер бачити звук подібно до людини, ми будуємо мел-фільтробанк — набір спеціальних фільтрів, кожен з яких пропускає лише певну смугу частот. Але на відміну від звичайної лінійної шкали, ці фільтри розташовані не рівномірно по герцах, а рівномірно по мелах.

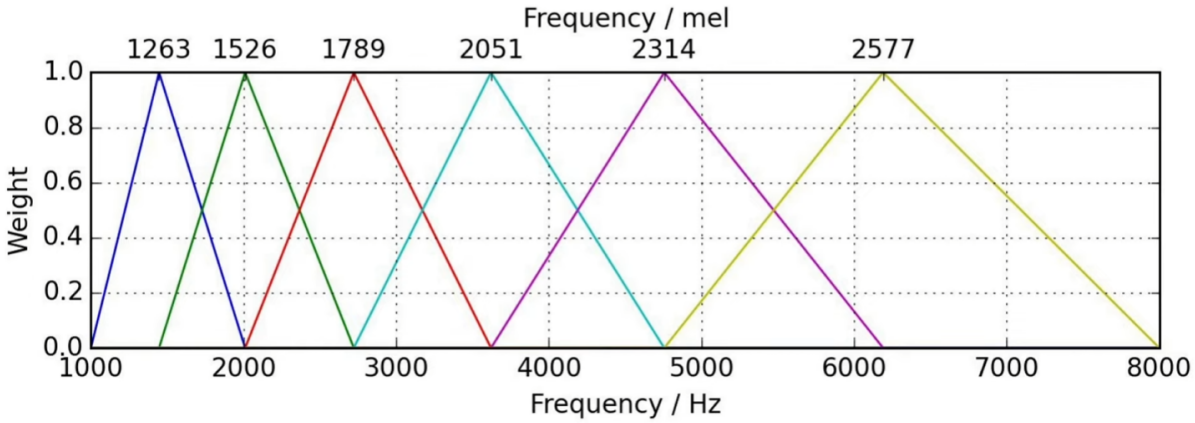
Це означає, що на низьких частотах (до 1 кГц) фільтри дуже густо розташовані — вони «розрізняють» тонкі зміни голосу або формант у мовленні, а на високих частотах (5–8 кГц) фільтри розміщені рідше — бо людське вухо все одно не бачить великої різниці між звуками, розділеними на кілька сотень герців у цьому діапазоні. Таким чином, мел-фільтробанк — це ніби «імітація внутрішнього вуха», де кожна ділянка реагує на свою групу частот, а загальний набір утворює природну шкалу нашого слуху.

3. Мел-фільтри: перцептивне групування частот у слухові смуги

Після перетворення частотної осі спектрограми у мел-шкалу постає питання: як об’єднати енергію сусідніх частот так, щоб це відповідало реальному сприйняттю слухом? Адже людина не сприймає кожну частоту окремо — слухова система аналізує звук у вигляді ширших частотних смуг, у яких кілька близьких частот об’єднуються в один «перцептивний канал». Саме для моделювання цього механізму використовують мел-фільтри — набір спеціальних вагових функцій $H_j[i]$, які визначають, як енергія різних частот об’єднується у вужчі або ширші діапазони відповідно до чутливості слуху.

Мел-фільтробанк — це система із кількох десятків (зазвичай 20–40) трикутних фільтрів, які покривають увесь частотний діапазон аналізу — наприклад, від 300 до 8000 Гц. Кожен фільтр відповідає окремій ділянці мел-шкали і моделює одну «слухову смугу». На низьких частотах такі фільтри розташовані густіше й перекриваються на невеликі інтервали, що відповідає високій роздільності слуху в цьому діапазоні. На високих частотах, навпаки, фільтри ширші й рідше розташовані — так, як людське вухо відрізняє звуки гірше.

Ідея полягає в тому, що кожен фільтр має найбільшу вагу (1.0) на своїй центральній частоті й поступово зменшує її до нуля на сусідніх межах. Таким чином, частота, яка лежить ближче до центру фільтра, робить більший внесок у його енергію.



Зображення було узято з *https://siggigue.github.io/pyfilterbank/melbank.html*

Процес створення мел-фільтрів є цілком формалізованим і базується на переході між шкалами герців та мелів. Задаються нижня та верхня межі аналізу — наприклад, f_{min} = 300 Гц, f_{max} = 8000 Гц. Ці частоти переводяться у мели за формулою m = 2595 · $\log_{10}$(1 + f / 700). Далі між мінімальним і максимальним значенням мелів рівномірно розподіляється задана кількість точок (наприклад, 40 + 2 допоміжні для меж). Отримаємо m_0, m_1, …, m_{n+1} рівновіддалені значення у мел-шкалі. Кожне з них знову перетворюється у герци за зворотною формулою f = 700 · (10^(m / 2595) − 1). аким чином ми дізнаємося, де саме на осі герців повинні розташовуватись центри та межі фільтрів. Для кожного фільтра j визначається ліва межа f_l, центр f_c і права межа f_r,. Потім створюється трикутна функція H_j[i]: між f_l і f_cзначення ростуть лінійно від 0 до 1; між f_c і f_r спадають від 1 до 0; поза цим діапазоном фільтр дорівнює нулю. Таким чином, мел-фільтри визначаються не з даних сигналу, а конструктивно — з теоретичних уявлень про те, як слух розподіляє частоти. Після побудови фільтрів кожен із них накладається на спектр сигналу, отриманий після короткочасного перетворення Фур'є. Якщо позначити енергію спектра у фреймі k як P_k[i], то енергія у j-тій мел-смузі обчислюється за формулою:

$$E_{j(k)} = \Sigma\,(P_k[i] \cdot H_j[i])$$

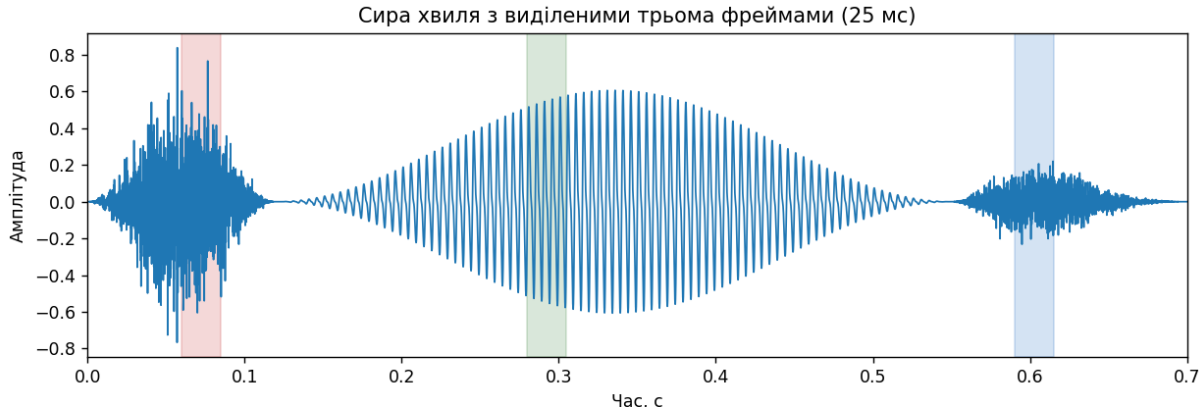
Тобто ми перемножуємо енергію кожної частоти на вагу, яку їй надає відповідний фільтр, і підсумовуємо результат. Отримані значення $E_{j(k)}$ показують, скільки енергії сигналу припадає на кожну перцептивну смугу. Для наближення до суб'єктивного відчуття гучності ці значення також перетворюються логарифмічно:

$$S_{j(k)} = log_{10}(E_{j(k)} + \varepsilon)$$

Після цього утворюється двовимірна матриця $S_{j(k)}$ — мел-спектрограма Мел-фільтри відіграють ключову роль у побудові мел-спектрограм та MFCC (Mel-Frequency Cepstral Coefficients) — найпоширенішого набору ознак у розпізнаванні мовлення. Вони дозволяють відкинути надлишкову деталізацію, зберігши при цьому саме те, що важливо для слухового сприйняття: форму спектра, розташування енергії між частотами та характер зміни тембру. У підсумку мел-фільтробанк можна вважати аналогом слухового аналізатора, що перетворює фізичну спектрограму на "перцептивну мапу", де енергія розподілена за слуховими смугами, подібно до того, як її обробляє людський мозок.

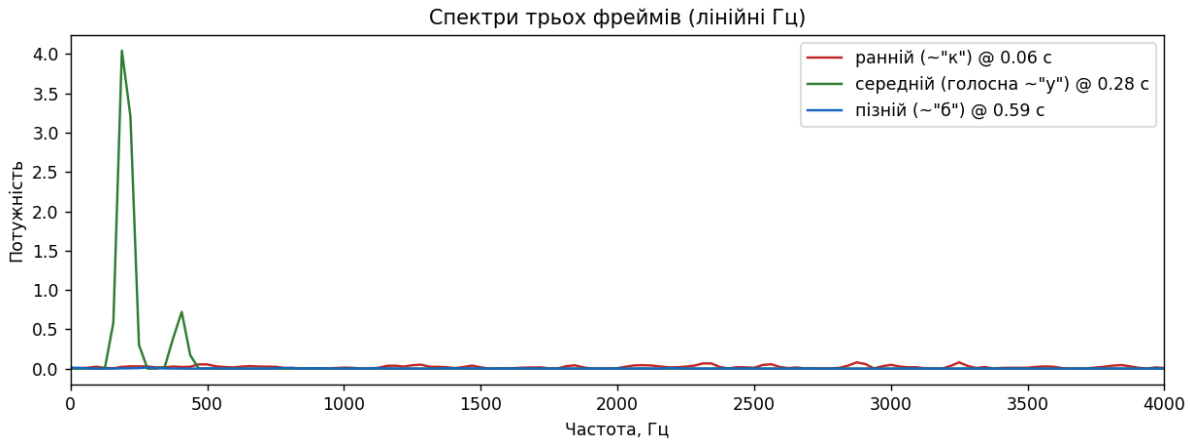
4. Мел-спектрограма: від хвилі до перцептивної «карти»

Уявімо, що в нас є сирий звуковий сигнал — наприклад, запис голосу або удар по барабану. На графіку він виглядає як хвиля, що коливається вздовж осі часу. Це просто зміна тиску повітря, яку мікрофон перетворив на цифрові відліки — числа, що показують, наскільки сильно коливається мембрана в кожен момент часу. Наприклад, ось короткий фрагмент звуку “куб”: сигнал спочатку має плавну форму (звук «к»), потім різкий імпульс (голосний «у»), і в кінці короткий спад (приголосний «б»).



Усе це — лише коливання амплітуди, які ми бачимо як хвилю, але слух розпізнає в ній смислові звуки.

Розкладемо цю хвилю на частоти, щоб побачити, які саме звуки (висоти, тони) утворюють сигнал у кожен момент часу.



Для аналізу таких сигналів недостатньо розглядати їх лише у часовій області — потрібно побачити, як енергія розподіляється за частотами у кожен момент часу. Саме для цього застосовують короткочасне перетворення Фур'є (Short-Time Fourier Transform, STFT). Воно дозволяє описати, які частоти «живуть» у сигналі зараз, тобто робить аналіз у двох вимірах — часі та частоті одночасно.

Щоб побачити, як змінюється спектр, ми ділимо сигнал на дуже маленькі шматочки — фрейми (зазвичай 20–30 мілісекунд). У межах даного відрізка звук встигає мало змінитися, тому його можна вважати «сталішим».

Далі для кожного фрейма виконується короткочасне перетворення Фур'є, яке дозволяє розкласти сигнал на набір гармонік, тобто показати, скільки енергії міститься у кожній частоті. Якщо звичайне дискретне перетворення Фур'є (DFT) показує частотний склад усього сигналу, то STFT дозволяє побачити, як цей склад змінюється з часом. Для кожного фрейма k обчислюється спектр:

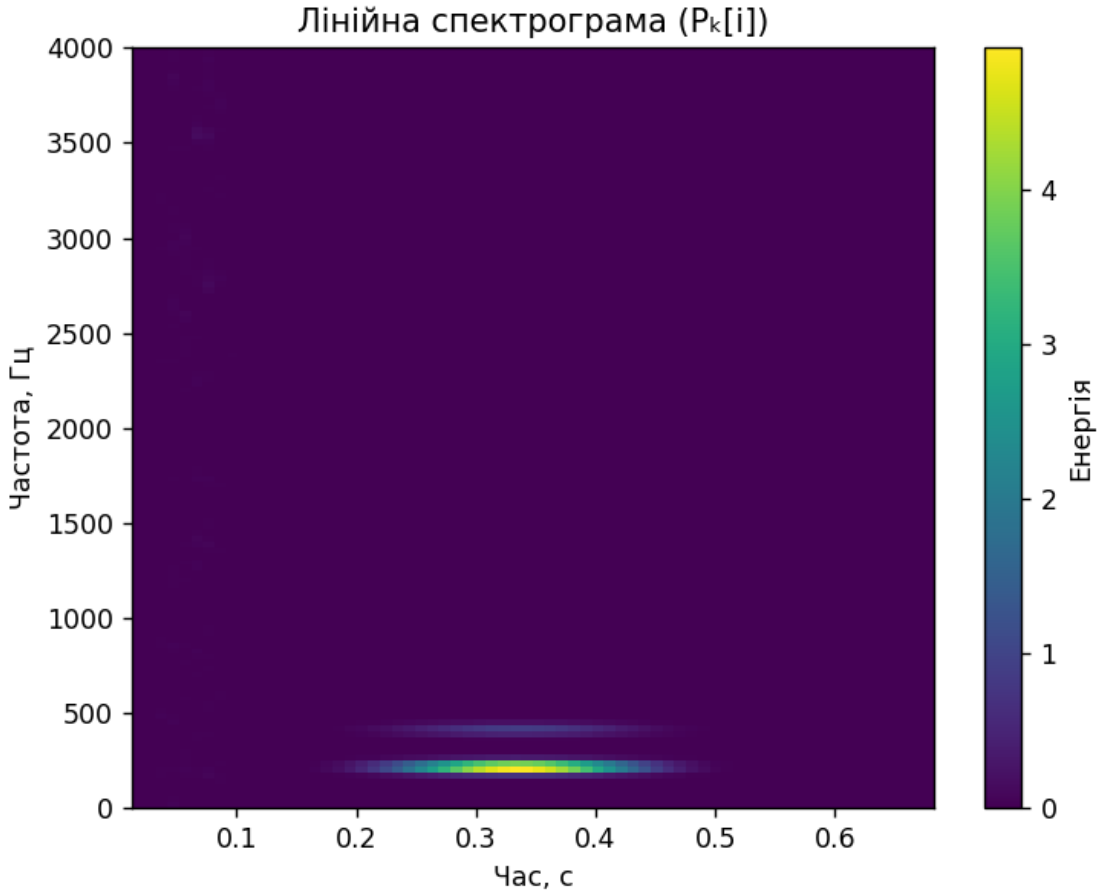
$$X_k[i] = \sum_{k=0}^{N/2-1} x_k \cdot w_{(k} - t_{0)} \cdot e\text{^}(-j \cdot 2\pi \cdot f_n \cdot k \,/\, f_s)$$

x_k — вхідний дискретний сигнал $w_{(k}-t_{0)}$ — віконна функція t_0 — часова позиція центра або початку вікна. $f_n = n \cdot f_s/N$). N — довжина вікна. f_s — частота дискретизації. $e\text{^}(-j \cdot 2\pi \cdot f_n \cdot k/F_s)$ — комплексна експонента.

Отримане $X_k[i]$ показує не лише частоту, а й фазу, але нас зазвичай цікавить лише енергія, яку ця частота несе. Тому обчислюють спектральну потужність (іноді кажуть «амплітудний спектр» або «енергетичний спектр»):

$$P_k[i] = (1 / N) \cdot |X_k[i]|^2$$

де $|X_k[i]|^2$ — квадрат модуля комплексного числа $X_k[i]$, який показує енергію даної частоти. Коли ми обчислили $P_k[i]$ для кожного фрейма, ми отримуємо набір «спектрів моментів часу». Якщо розташувати їх поруч у часовій послідовності, утворюється спектрограма — своєрідна двовимірна карта, де по осі X відкладається час (номер фрейма), по осі Y — частота (Гц), колір або яскравість кожної точки показує енергію сигналу.

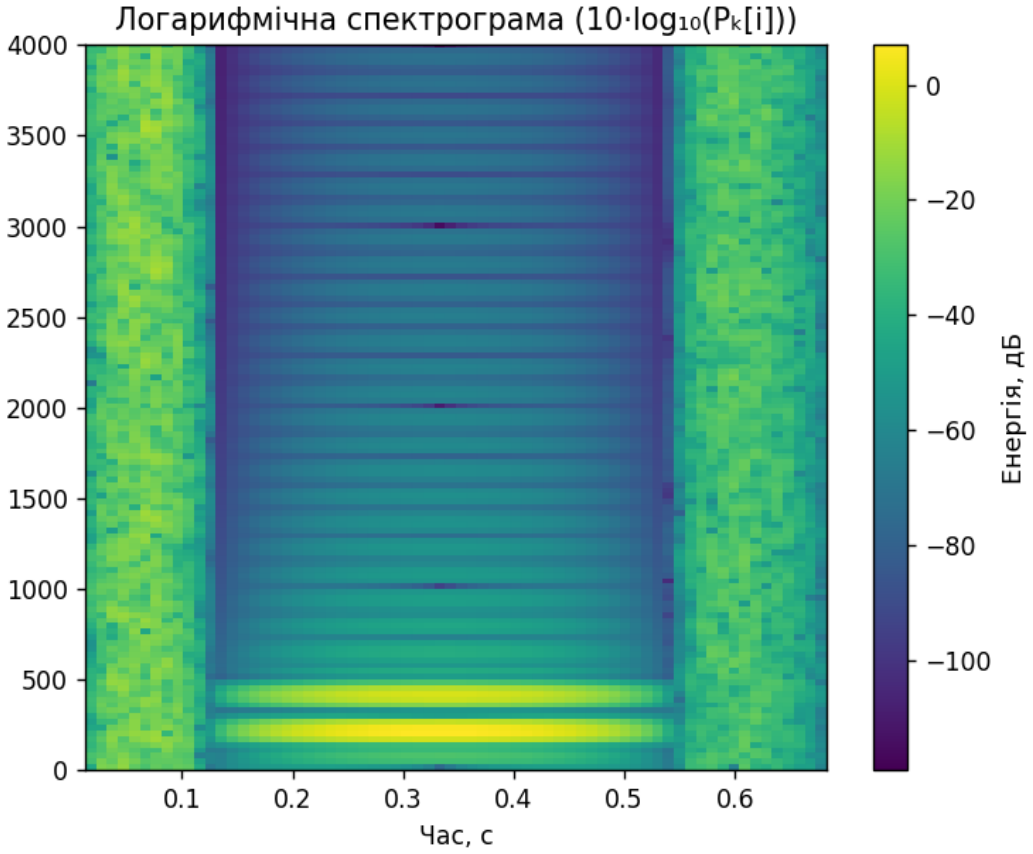


Спектрограма містить повну інформацію про частотний склад сигналу, однак побудована у фізичних одиницях — герцах і лінійній енергетичній шкалі. Це робить її малопридатною для подальшого аналізу мовлення або інших звуків, оскільки вона не враховує особливості людського слуху. На такій спектрограмі потужні частоти домінують, пригнічуючи слабші компоненти, а частотна шкала не відповідає сприйняттю висоти тону. Тому лінійна спектрограма використовується лише як проміжний етап.

Людський слух сприймає зміни гучності не лінійно. Якщо інтенсивність звуку збільшити у десять разів, ми не відчуваємо його у десять разів голоснішим — лише приблизно вдвічі. Цю властивість описує закон Вебера–Фехнера, згідно з яким суб'єктивне відчуття сили стимулу пропорційне логарифму його фізичної інтенсивності. Тому для звукових сигналів використовують логарифмічне перетворення енергії, яке стискає діапазон значень і робить різницю між тихими та гучними ділянками ближчою до того, як ми їх реально чуємо. Таке перетворення описується формулою:

$$P_k(dB)[i] = 10 \cdot \log_{10}(P_k[i] + \varepsilon).$$

де ε — мале число, що запобігає обчисленню log(0). Логарифмічне подання (у дБ) дозволяє виділити слабкі частотні компоненти та є стандартною формою представлення спектрограм у задачах розпізнавання мовлення.



Ми отримуємо картину, яка ближче до того, як вухо “бачить” звук, а не як його вимірює прилад. На спектрограмі вимови “куб” у першій частині видно широку смугу високочастотного шуму (приголосна «к»), посередині — чіткі горизонтальні лінії (гармоніки голосної «у»), а наприкінці — різке згасання енергії (приголосна «б»). Таким чином спектрограма показує, які частоти присутні у сигналі і як їх інтенсивність змінюється з часом. Це дозволяє побачити структуру мовлення, музики або будь-якого іншого звуку у вигляді наочної «карти».

Однак логарифмування енергії — це лише половина перцептивного узгодження. Людське вухо сприймає не тільки зміни гучності нелінійно, а й висоту тону — тобто різницю між частотами. Для нас різниця між 200 Гц і 300 Гц відчувається значно сильніше, ніж між 5200 Гц і 5300 Гц, навіть якщо фізично це той самий крок у 100 Гц. Щоб урахувати й цю особливість, саму частотну вісь також перетворюють у нелінійну шкалу — мел-шкалу. У ній низькі частоти розтягнуті (вища роздільність), а високі — стиснуті. Це вже наступний крок після логарифмування енергії: він формує основу для побудови мел-спектрограми, де обидві сторони — і енергія, і частота — узгоджені з особливостями людського слуху. Перехід із частот (Гц) до мелів описується емпіричною формулою:

$$m = 2595 \cdot \log_{10}(1 + f / 700)$$

а зворотне перетворення:

$$f = 700 \cdot (10^{(m / 2595)} - 1)$$

де m — значення у мелах, а f — фізична частота в герцах. Після перетворення шкали частот ми будуємо набір мел-фільтрів — це трикутні вікна, розташовані рівномірно по мел-ось, а не по герцах. Їхній принцип такий: кожен фільтр $H_j[i]$ «підсумовує» енергію у певному мел-діапазоні, а

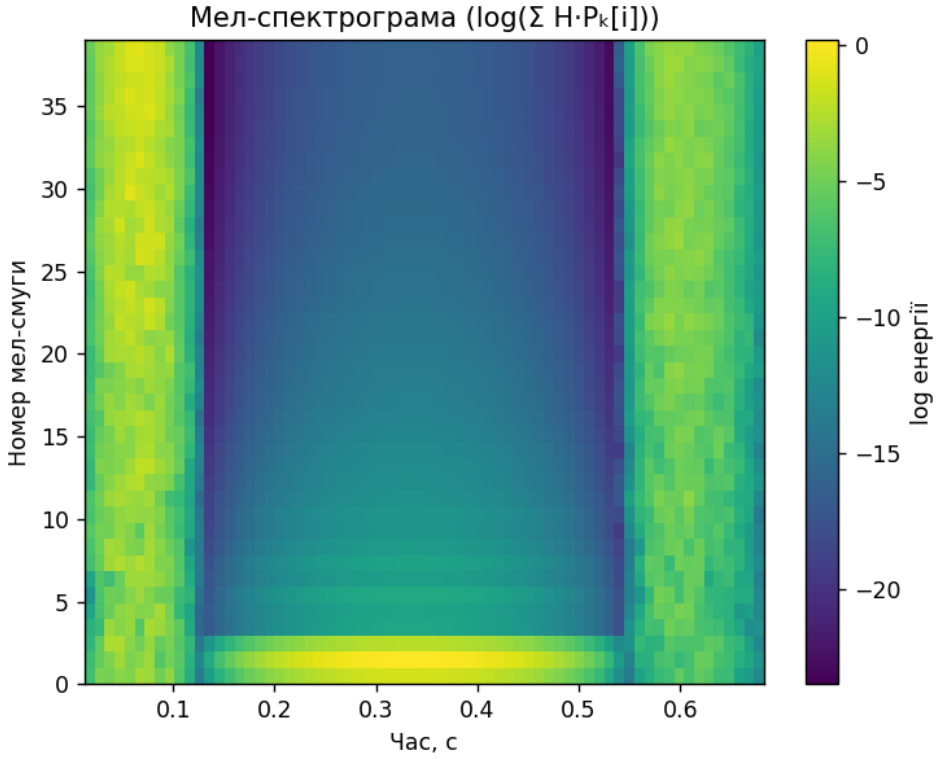
сусідні фільтри частково перекриваються, щоб не втратити плавності. Кількість фільтрів зазвичай 20–40 — залежно від бажаної роздільності. Формально енергія $E_{j(k)}$ у j-му фільтрі для фрейма k обчислюється як:

$$E_{j(k)} = \Sigma\,(P_k[i] \cdot H_j[i])$$

Де $P_k[i]$ — енергія частоти *i* у фреймі k, а $H_j[i]$ — вага відповідного фільтра. Після цього отримані енергії також логарифмують, щоб узгодити гучність:

$$S_{j(k)} = log_{10}(E_{j(k)} + \varepsilon)$$

Отримана матриця $S_{j(k)}$ — це і є мел-спектрограма, де по осі j — номери мел-смуг (перцептивні частоти), по осі k — час (номер фрейма), а колір відображає енергію у логарифмічній шкалі.



Таким чином, логарифмування енергії забезпечує слухову динамічну компресію, а мел-шкала — перцептивне вирівнювання частотної осі. Разом ці два кроки утворюють основу для побудови мел-спектрограми —

наступного рівня аналізу, який відображає не лише фізичні, а й психоакустичні властивості звуку.

Усі попередні етапи — короткочасне перетворення Фур'є, логарифмування енергії та перехід до мел-шкали — виконуються не просто як математичні маніпуляції. Їх головна мета — наблизити цифрове представлення звуку до того, як його сприймає людина. Звичайний звуковий сигнал у часовій формі містить мільйони відліків, але для систем автоматичного розпізнавання мовлення чи класифікації звуків таке представлення надлишкове і складне для аналізу. Нейронна мережа не може ефективно навчитися з «сирих» хвиль, якщо не надати їй характеристик, що мають сенс із точки зору сприйняття. Саме тут і з'являється мел-спектрограма — двовимірна карта, де по осі X відкладено час, а по осі Y — частотні смуги, узгоджені з мел-шкалою. Колір або яскравість кожної точки відображає логарифм енергії в цій смузі у даний момент часу. У такій формі звук перетворюється на своєрідне «зображення», де вертикальні структури відповідають шумам або приголосним, а горизонтальні — голосним або сталим тоном. Нейронні мережі, особливо згорткові (CNN) або рекурентні (RNN/LSTM), дуже добре навчаються розпізнавати саме такі візерунки: комбінації частотних смуг, їхню динаміку та характер змін у часі. Мел-спектрограма тому і стала стандартом у сучасних системах розпізнавання мовлення, класифікації музики, виявлення емоцій та акустичних подій — вона одночасно зберігає фізичну точність сигналу і враховує фізіологічні особливості слуху.

5. Мел-частотні кепстральні коефіцієнти (MFCC)

Після отримання мел-спектрограми логічним наступним кроком є перехід до ще більш компактного й інформативного представлення звуку — мел-частотних кепстральних коефіцієнтів (Mel-Frequency Cepstral Coefficients, MFCC).

Вони стали одним із найважливіших стандартів у цифровій обробці мовлення, музики та аудіосигналів загалом, оскільки дозволяють зменшити обсяг даних, зберігаючи при цьому найважливіші характеристики спектра, пов’язані з тембром, артикуляцією та формантною структурою звуку. Термін кепстр (cepstrum) походить від слова спектр, записаного навпаки, і натякає на те, що ми проводимо аналіз спектра спектра.

SPECtrum => **CEPS**trum

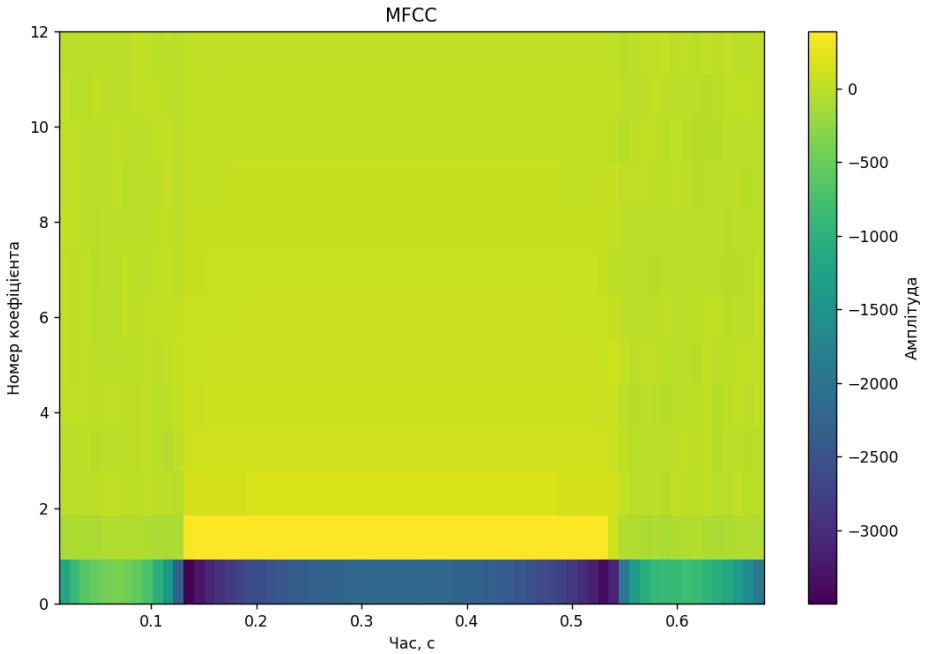
Якщо спектр показує розподіл енергії сигналу за частотами, то кепстр дає змогу дослідити, як змінюється форма цього спектра — тобто визначає більш повільні коливання амплітуди по частоті, які відповідають резонансам мовного тракту (формантам). Іншими словами, MFCC відокремлюють високочастотну структуру спектра (джерело — голосові зв’язки) від низькочастотної обвідної (фільтр — ротова порожнина). Саме це розділення лежить у серці моделі “джерело + фільтр”, що використовується в усіх теоріях мовлення.

Мел-спектрограма все ще має надлишкову інформацію. Сусідні мел-смуги перекриваються, тому значення їх енергій сильно корельовані між собою: якщо в одній смузі зростає енергія, у сусідній вона, як правило, також підвищується. Для того щоб отримати більш незалежні параметри, використовують дискретне косинусне перетворення (DCT) — математичну операцію, яка дозволяє розкласти послідовність логарифмованих енергій на суму косинусів різних частот. Косинусне перетворення виконує роль “компресора інформації”: воно переносить енергію з великої кількості

сильно пов’язаних змінних (енергій у мел-смугах) у невелику кількість незалежних коефіцієнтів, які описують загальну форму спектра. Формально DCT для кожного моменту часу *k* записується як:

$$C_{n(k)} = \Sigma\, S_{j(k)} \cdot cos[\ (\pi\, n\, /\, M) \cdot (j - 0.5)\ ]$$

де $C_{n(k)}$ — n-й кепстральний коефіцієнт для фрейма k; $S_{j(k)}$ — логарифм енергії j-тої мел-смуги; M — кількість мел-смуг (наприклад, 40); n — номер коефіцієнта (зазвичай 0–12). Сенс цієї формули полягає в тому, що ми описуємо форму спектра за допомогою коливальних базисних функцій (косинусів). Перші кілька таких функцій змінюються повільно — вони описують загальний нахил і вигин спектра. Наступні, з більшою “частотою”, описують дрібні деталі. На практиці виявляється, що більшість важливої інформації про тембр міститься у перших 12–13 коефіцієнтах, тому інші можна просто відкинути. Це зменшує розмірність даних у декілька разів без суттєвої втрати змісту.



На побудованому графіку MFCC ми бачимо, що перші коефіцієнти мають великі, чітко виражені значення, а далі амплітуда поступово зменшується і

стає майже непомітною. Це не випадковість, а наслідок того, як побудована DCT. У лог-мел спектрі більшість енергії зосереджена у низьких “косинусних частотах”, тобто у повільних змінах спектра. Такі зміни описуються саме першими кількома коефіцієнтами — вони мають великі значення, тому що на них припадає основна енергетика мовного сигналу. Вищі коефіцієнти, що відповідають за дрібні деталі, отримують усе менше енергії — тому на графіку вони виглядають тьмяними або майже зникають. Фактично, ми отримуємо ефект, подібний до низькочастотного фільтра: DCT концентрує найважливішу інформацію внизу (у перших компонентах), а незначні варіації — у високих.

C_0 — середня енергія сигналу, тобто його “гучність”. C_1–C_2 — описують нахил спектра (наскільки енергія зосереджена в низьких або високих частотах). Вони відображають форму ротової порожнини, положення язика, губ, тобто те, що визначає вимову. C_3–C_6 — уточнюють розподіл енергії між окремими формантами, відповідають за характерні темброві особливості звуку. C_7 і вище — дрібні шумові деталі, мікровібрації або залишкові відлуння; часто не мають вагомого впливу на розпізнавання мовлення, тому відкидаються.

Таким чином, перші коефіцієнти MFCC концентрують основну фонетичну інформацію про звук, а решта несе лише малозначущі деталі. Саме тому в більшості систем розпізнавання мовлення беруть лише 12–13 перших коефіцієнтів — цього достатньо, щоб нейронна мережа навчилась розрізняти голосні, приголосні й інтонаційні відтінки.

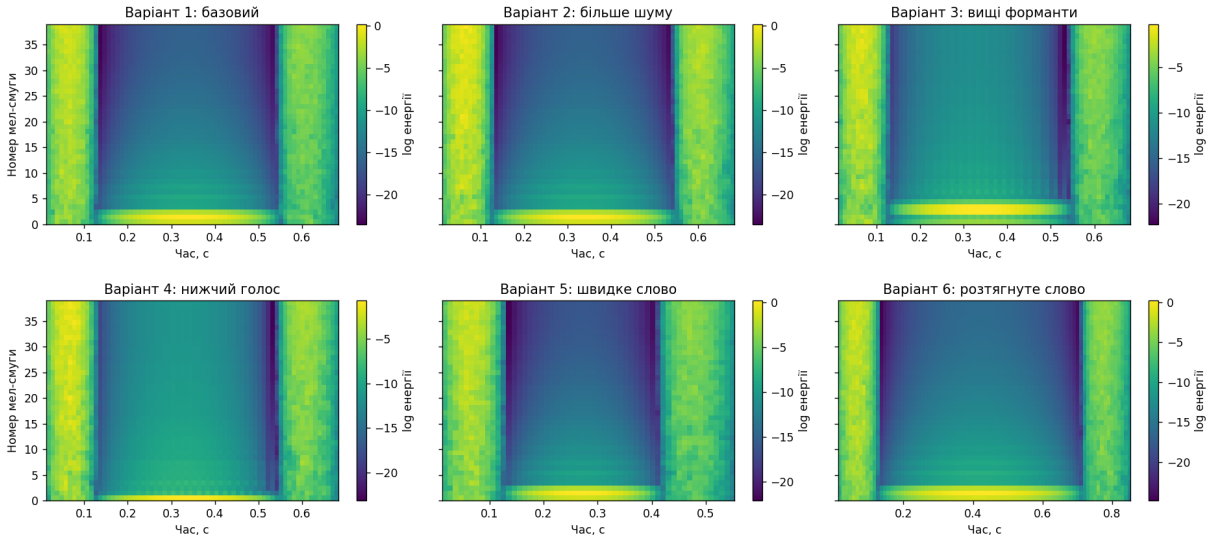
Після обчислення мел-частотних кепстральних коефіцієнтів (MFCC) ми отримуємо компактне, інформативне й перцептивно узгоджене представлення звукового сигналу. Наступним етапом є використання цих

ознак як вхідних даних для моделей машинного навчання. На сьогодні найбільш ефективними для задач розпізнавання мовлення є різні архітектури нейронних мереж. MFCC у цьому контексті виконують роль «мовних пікселів» — атомарних ознак, з яких нейромережа будує розуміння мовного сигналу. Сирий аудіосигнал має надто високу розмірність. Наприклад, 1 секунда звуку з частотою дискретизації 16 кГц містить 16 000 чисел і навіть просте речення може складатися з сотень тисяч відліків. Передавати сирий сигнал прямо у нейронну мережу надзвичайно обчислювально дорого, містить багато зайвої інформації, складно для навчання — мережа не може автоматично виділити тембр, форманти, шум тощо, чутливо до якості мікрофона, гучності, луни. MFCC перетворюють цей складний сигнал у компактний набір параметрів, які вже містять структурну та перцептивно значущу інформацію. Зазвичай один фрейм звуку описується 12 або 13 MFCC-коефіцієнтами, що зменшує розмірність ознак у 1000 разів.

РОЗДІЛ ДРУГИЙ

Навчання нейронної мережі

Процес навчання нейронної мережі для розпізнавання голосових команд базується на перетворенні акустичного сигналу у більш зручне для машинної обробки представлення, яке водночас зберігає ключові властивості людського мовлення. Найбільш поширеною формою такого представлення є мел-спектрограма, що відображає розподіл енергії сигналу по частотах, перелічених у нерівномірній шкалі Мела. Мел-шкала є психоакустичною, тобто такою, що узгоджується з тим, як людське вухо сприймає висоту звуку. Саме тому перехід від звичайної спектрограми до мел-спектрограми дозволяє виділити ознаки, які є максимально значущими з точки зору реального людського слухового аналізатора. Навчання моделі починається з формування великої та різноманітної колекції звукових прикладів, у яких одне й те саме слово або команда вимовлені багатьма людьми, у різних умовах, з різною гучністю, тембром, швидкістю та ступенем шумової завади. Це необхідно для того, щоб модель навчилася розпізнавати не окремі акустичні зразки, а узагальнені закономірності, характерні саме для конкретного слова.



6 варіантів промови одного слова “куб”

Кожен аудіосигнал проходить однаковий ланцюг попередньої обробки: нормалізацію, сегментування на короткі вікна, обчислення короткочасного перетворення Фур'є, застосування мел-фільтробанку та логарифмування енергетичних значень. У результаті кожне слово перетворюється на двовимірну матрицю – мел-спектрограму, де вісь часу відображає динаміку слова, а вісь частот – його гармонічний зміст.

Після цього мел-спектрограми подаються в нейронну мережу, яка у процесі навчання повинна знайти закономірності, що повторюються у всіх зразках цільового слова, і водночас навчитися ігнорувати другорядні варіації: шум, зміни гучності, коливання висоти голосу, індивідуальні особливості мовця. У найпростішій моделі мережа працює як складний нелінійний перетворювач, який обчислює для кожної вхідної матриці певний набір ознак – спочатку низькорівневих (локальні контури спектра, перепади енергії, формантні піки), а на глибших шарах – більш абстрактних.

Навчання побудоване на багаторазовому прогоні всіх даних через мережу. На початковому етапі ваги мережі не мають значення і моделюють практично випадкові перетворення. Тому перші передбачення є хаотичними та неправильними. Але завдяки так званому прямому проходу мережа обчислює поточний прогноз, після чого порівнює його з реальним правильним значенням. Різниця між передбаченням та істинною міткою формує величину помилки, яка в рамках математичної моделі називається функцією втрат. Саме ця помилка і є керуючим сигналом, на який орієнтується вся подальша процедура корекції.

Під час зворотного поширення помилки алгоритм крок за кроком обчислює, як кожен параметр мережі вплинув на неточність передбачення, і відповідно коригує цей параметр у напрямку зменшення втрат. Умовно кажучи, мережа поступово «вчиться» правильніше реагувати на ті частотні

структури мел-спектрограми, які є типовими для слова, що вивчається. Якщо це слово має характерний шумовий вибух на початку, мережа навчиться розпізнавати такий акустичний малюнок. Якщо слово містить специфічне розташування формант, модель знайде відповідний патерн у спектральному просторі. Якщо амплітудна структура слова змінюється у певний спосіб, мережа теж буде враховувати це у своєму рішенні.

Процес навчання повторюється багаторазово: кожен цикл, що складається з прямого проходу, обчислення втрат та зворотного поширення помилки, називається епохою. Зазвичай потрібні десятки або навіть сотні епох, поки модель не починає стабільно розпізнавати внутрішні закономірності у даних. Кожна нова епоха поступово «стискає» спектр можливих помилкових станів моделі, концентруючи її параметри навколо тих значень, які найточніше відтворюють правильну відповідь. У цьому сенсі нейронна мережа поводиться як функція, яка адаптується під усе більш складні деталі структури мовлення, водночас зберігаючи здатність узагальнювати.

Важливо, що алгоритм навчання забезпечує стійкість до шуму та різноманітних спотворень. Оскільки у тренувальному наборі містяться приклади різних умов запису, мережа навчається не прив’язуватися до конкретного набору акустичних особливостей, а концентруватися на фундаментальних структурах, притаманних самому слову. Наприклад, навіть якщо слово «куб» вимовляється у галасливому середовищі, мережа, що достатньо натренована на великій вибірці, залишатиметься здатною викривати характерний набір мел-енергій та час-частотних ознак, що ідентифікують це слово.

У кінцевому підсумку модель переходить від етапу навчання до етапу інференсу – тобто до режиму розпізнавання. На цьому етапі мережа вже не змінює своїх параметрів: вона застосовує набуті під час тренування знання

для нових звукозаписів. Користувач промовляє слово, аудіосистема будує його мел-спектрограму, і мережа одразу видає ймовірність того, що вимовлене слово відповідає цільовій команді. У практичних застосуваннях, наприклад у вбудованих системах або мікроконтролерах, ця ймовірність порівнюється з певним порогом, після чого система активує відповідну дію.

Таким чином, навчання нейронної мережі для голосового розпізнавання на основі мел-спектрограм є складним процесом, що поєднує математичні методи аналізу сигналів, психоакустичні принципи розподілу частот та глибокі методи оптимізації. В основі лежить ідея про те, що голос можна описати у вигляді структурованого зображення, а складні акустичні структури можна навчити розпізнавати тим самим методом, яким розпізнають обличчя, літери, жести або природні сцени. Саме ця уніфікація дозволяє сучасним моделям досягати високої точності розпізнавання голосових команд навіть на обмежених обчислювальних ресурсах, таких як ESP32-S3 або інші вбудовані процесори.

Навчання фонем

Розпізнавання мовлення через фонеми ґрунтується на принципово іншому підході, ніж розпізнавання окремих слів як неподільних акустичних шаблонів. У класичних системах командного розпізнавання кожне слово тренується окремо, і модель фактично вивчає характерну мел-спектрограму конкретного звукового патерну. Однак у природній мові слова складаються з менших одиниць – фонем, тобто мінімальних звукових елементів, які мають власну акустичну структуру і відрізняються тим, як вони формуються мовним апаратом. Фонемний підхід робить процес розпізнавання значно гнучкішим, бо замість того, щоб окремо вчити модель на тисячі слів, достатньо навчити її правильно розпізнавати порівняно невелику кількість фонем і правила їх поєднання.

У цьому підході першим етапом, як і в інших системах, є перетворення сигналу у час-частотну форму: спектрограму або мел-спектрограму. Проте подальша інтерпретація такої матриці вже орієнтована не на категорію «слово», а на категорію «фонема». Модель повинна навчитися бачити у мел-спектрограмі характерні патерни, що відповідають голосним, приголосним, вибуховим, фрикативним або сонорним звукам. Наприклад, голосні утворюються відкритою голосовою щілиною, тому їх спектр насичений стабільними формантами – горизонтальними смугами підвищеної енергії. Приголосні фрикативні звуки мають шумовий спектр із підвищеною енергією у верхніх частотах. Вибухові приголосні характеризуються короткими імпульсами енергії або різкими переходами. Усе це формує складну систему закономірностей, яку можна побачити лише в час-частотному поданні.

Навчання на фонеми відбувається за класичною схемою нейронних мереж глибокого навчання, але з чітким акцентом на класифікацію кожного

короткого фрейму. У типовому сценарії кожен фрейм мел-спектрограми (наприклад, 25–30 мс) асоціюється з певною фонемою, яка була вимовлена у цей момент. Однак реальні мовні процеси плавні: звуки накладаються один на одного, голосові апарати не перемикаються між фонемами миттєво, і тому реальна фонема зазвичай простягається на декілька сусідніх фреймів. Це створює ситуацію, коли задача перетворюється на послідовну класифікацію, тобто модель повинна враховувати контекст навколишніх кадрів, щоб правильно визначити фонему.

Для цього найчастіше застосовують дві архітектури: згорткові нейронні мережі, які виділяють локальні акустичні особливості у спектрограмі, та рекурентні або трансформерні мережі, що вміють враховувати часові залежності. Згорткові шари чудово виявляють характерні форми хвиль, структури формант, частотні переходи або шумові компоненти. Рекурентні шари (LSTM або GRU) розглядають фрейми як послідовність, «пам’ятаючи» інформацію про попередні стани артикуляції. Таким чином модель отримує той тип контексту, який людина використовує природним шляхом: ми впізнаємо фонему не лише за тим, що звучить у цю мить, а й за тим, як до цього звучали сусідні звуки.

У процесі навчання фонемної моделі кожна мел-спектрограма розбивається на послідовність фреймів, і для кожного з них визначається правильна фонема. Це створює великий масив пар «вхідний фрейм → правильна фонема», на основі якого мережею виконується оптимізація. Під час прямого проходу модель намагається спрогнозувати фонему для кожного моменту часу. Потім це передбачення порівнюється з реальними мітками, і обчислюється функція втрат, яка показує, наскільки сильно мережа помиляється. На етапі зворотного поширення помилка розподіляється назад через усі шари моделі, і ваги коригуються так, щоб зменшити сумарну невідповідність.

Однак навчання фонемної моделі має свою особливість: мовлення не має точного, детермінованого поділу на фонеми, і різні люди вимовляють ті самі звуки по-різному. Крім того, фонеми залежать від контексту, що створюється сусідніми звуками: те саме «к» звучить інакше у словах «куб», «кіт» або «склад». Ці складнощі вирішуються завдяки тому, що модель навчається не на одному зразку, а на великій кількості прикладів, що охоплюють різні голоси, варіанти вимови, інтонації та швидкості мовлення.

Після того, як модель навчається стабільно розпізнавати фонеми, система переходить до етапу побудови більш високорівневих одиниць – слів. Це виконується за допомогою мовної моделі (language model) або послідовних алгоритмів, таких як CTC-декодування. Їхня мета полягає в тому, щоб взяти послідовність фонем і реконструювати з неї найбільш імовірне слово. Мовна модель ураховує всі можливі правила та статистичні закономірності поєднання фонем у словах. Наприклад, якщо модель розпізнала послідовність звуків, які наближено відповідають фонемам /k/-/u/-/b/, мовна модель підтвердить, що саме таке поєднання є валідним у конкретній мові, та з високою ймовірністю визнає, що було вимовлено слово «куб».

Таким чином, розпізнавання мовлення через фонеми перетворює задачу класифікації цілих слів у задачу розпізнавання значно дрібніших і фундаментальніших одиниць мови. Це робить систему набагато стійкішою, гнучкішою і здатною до масштабування. Модель, яка вміє впевнено розпізнавати фонеми, легко адаптується до нових слів, навіть тих, які не входили до тренувального набору, оскільки слова складаються з обмеженої кількості фонем. Такий підхід ближчий до того, як людина сприймає мовлення: ми чуємо не стільки «конкретні спектрограми», скільки характерні звуки, які утворюють слова, і саме на цьому рівні локалізуються ключові одиниці змісту.

Огляд досліджуваного прототипу розпізнавання голосових команд.

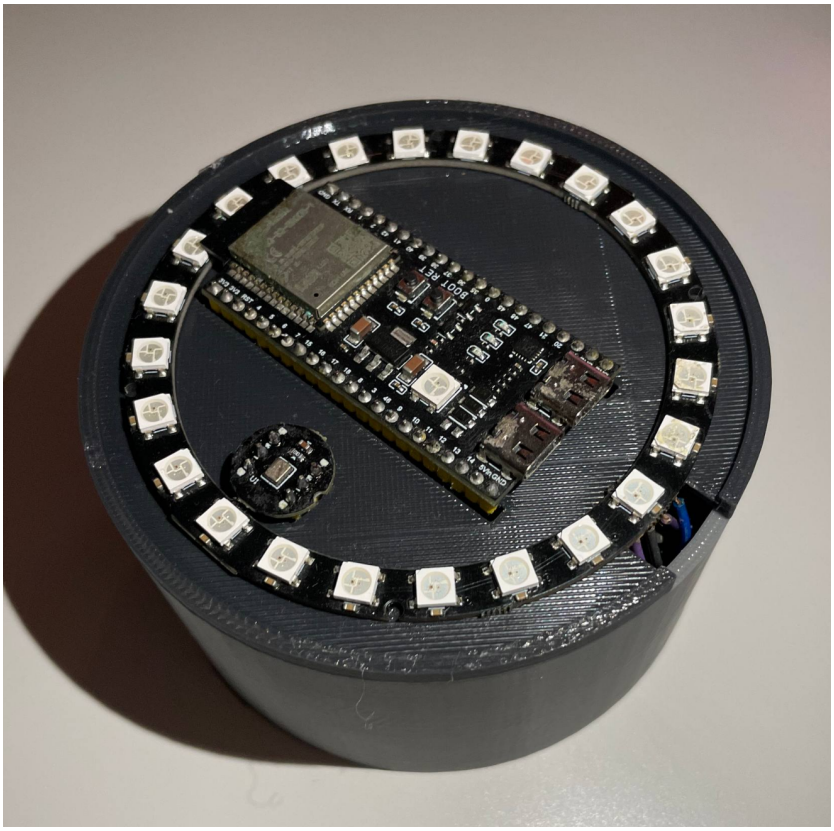
Пристрій являє собою компактну модульну апаратну систему, зібрану у циліндричному корпусі, що виготовлений переважно методом 3D-друку. Корпус забезпечує фіксацію всіх елементів, захист електроніки та формує акуратну зовнішню форму пристрою. Верхня панель конструкції виконує роль монтажної платформи для основних компонентів.

У центральній частині верхньої панелі розміщена плата мікроконтролера типу ESP32-S3 DevKit. Саме ця плата забезпечує обчислювальні ресурси, обробку аудіо та керування периферією. Вона ж містить бездротові модулі, живлення та цифрові інтерфейси.

Навколо контролера встановлене кільце адресних світлодіодів WS2812, розташоване по периферії конструкції. Кільце складається з окремих 24-ох світлодіодів, об’єднаних у послідовний цифровий ланцюг.

Підключення здійснюється через один лінійний цифровий сигнал DATA, що надходить від мікроконтролера. Живлення кільця подається від 3.3 В, а земля об’єднана з основною платою. Кільце використовується як основний візуальний інтерфейс і може відображати різні кольорові анімації.

Окремим важливим компонентом є цифровий MEMS-мікрофон MS3625, закріплений на верхній частині корпусу поруч із основною платою. MS3625 — це компактний всеспрямований мікрофон із цифровим інтерфейсом I2S, що дозволяє подавати аудіосигнал безпосередньо до мікроконтролера без зовнішніх аналогових підсилювачів та АЦП. Модуль мікрофона забезпечує високу чутливість і низький рівень шуму, що є критично важливим для роботи системи розпізнавання голосу. Його розташування у верхній частині корпусу мінімізує акустичні перешкоди та забезпечує прямий доступ до звукового поля.



Живлення пристрою здійснюється через USB-роз’єм або від зовнішнього блока живлення. Корпус також приховує всю комутацію, залишаючи доступними лише необхідні елементи.

Таким чином, конструкція пристрою складається з трьох основних апаратних блоків: мікроконтролер ESP32-S3, адресне світлодіодне кільце WS2812 та цифровий мікрофон MS3625, змонтованих у цілісному 3D-друкованому корпусі.

Огляд архітектури та логіки роботи проєкту.

Розроблений програмний комплекс являє собою модульну багатокомпонентну систему, побудовану відповідно до парадигми, закладеної в ESP-IDF. Проєкт структурується навколо ідеї чіткого розподілу функціональності між незалежними компонентами, кожен із яких реалізує свою частину логіки: від обробки голосових команд до візуалізації подій на світлодіодній стрічці. Такий підхід забезпечує не лише простоту супроводу та розширення, але й дозволяє досягти стабільної роботи в умовах реального часу, характерних для вбудованих систем.

Загальний вхід до програми реалізовано у файлі main/main.c, який, згідно з класичними принципами embedded-розробки, залишається максимально мінімалістичним. Тут відбувається лише первинний запуск двох ключових підсистем: обробки мови та графічного інтерфейсу користувача. Основне виконання не перевантажене додатковою логікою, оскільки обробка даних, приймання аудіо, збудження моделей глибинного навчання та відтворення візуальних патернів перенесено в окремі FreeRTOS-задачі, що живуть у власних компонентах. Така конструкція нагадує багатошарову систему, де головний потік виступає лише в ролі координатора, тоді як реальна робота відбувається всередині добре структурованої мережі модулів.

Центральним елементом проєкту є компонент sr, який відповідає за весь аудіо-та ML-тракт. Його ініціалізація відбувається через спеціалізований модуль sr_handler, який піднімає підсистему AFE (Audio Front-End) — ключовий компонент фреймворку esp-sr. У цьому процесі відбувається завантаження мо-дельних файлів із спеціальної моделі-партиції, конфігурація параметрів передобробки сигналу, підключення драйверів платформи через шар BSP та створення AFE-хендла, який інкапсулює всі аудіоалгоритми. У такий спосіб уся складність роботи з кодеками,

мікрофонною матрицею, I2S-інтерфейсом абстрагований рівень, доступний для використання вищими шарами.

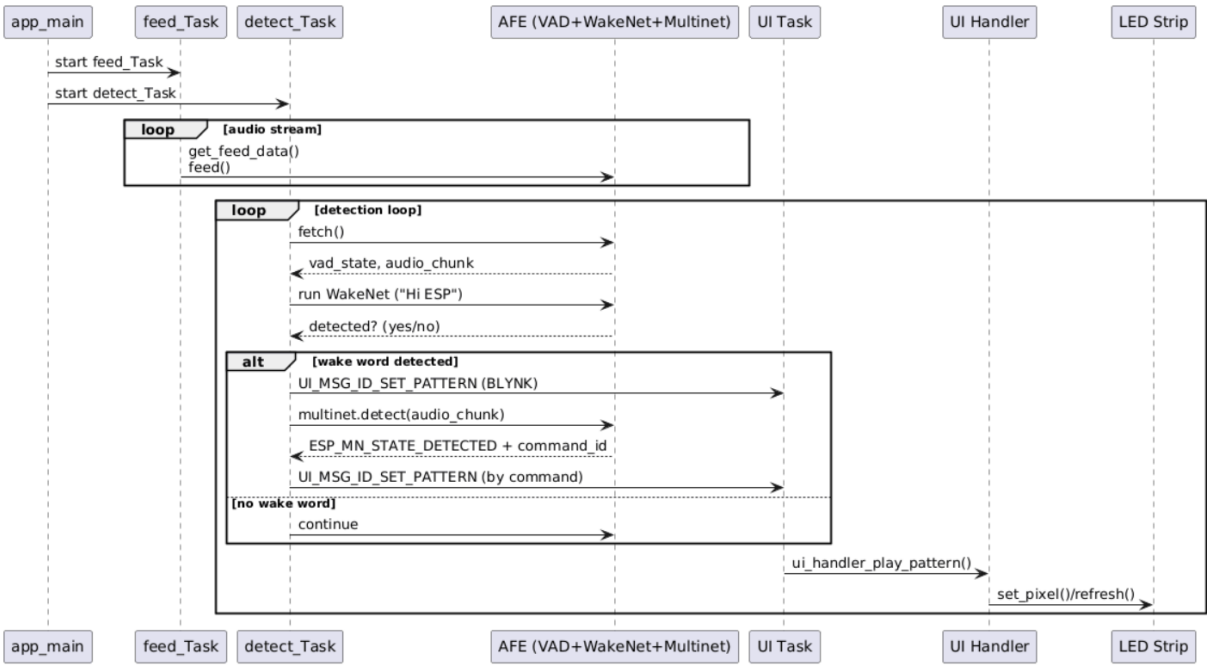
Особливістю архітектури є те, що аудіообробка рознесена на два паралельні завдання, що виконуються на різних ядрах ESP32-S3. Перша задача, так звана feed_Task, безперервно зчитує сирі аудіодані з мікрофонного інтерфейсу та передає їх до AFE із мінімальною затримкою. Друга задача — detect_Task — займається безпосереднім запуском моделі Multinet, аналізом мовного потоку та визначенням команд користувача. Такий поділ відповідає сучасним вимогам до побудови голосових інтерфейсів на мікроконтролерах і дозволяє гарантувати стабільність роботи навіть у ситуаціях підвищеного навантаження, коли затримка між надходженням фреймів аудіо та їхнім аналізом має бути мінімальною.

Визнані моделлю командні індекси мапляться на відповідні події графічного інтерфейсу, після чого передаються до UI-модуля через внутрішню чергу повідомлень. Таким чином створюється класична модель “producer – consumer”: аудіотракт генерує події, а користувацький інтерфейс їх асинхронно обробляє.

Другим великим компонентом є UI-підсистема, яка відповідає за візуальне представлення реакції пристрою. На апаратному рівні вона працює зі світлодіодною стрічкою WS2812, керованою через RMT-блок ESP32-S3. UI-модуль складається з багаторівневої структури: зовнішній API, менеджер задач, обробник патернів та низькорівневий драйвер. Задача UI блокується на очікуванні повідомлень, що надходять із SR компонента, а отримавши команду, відтворює відповідний патерн — плавну анімацію “хвоста”, кольоровий спалах або повне очищення стрічки. Цей підхід також втілює принцип розділення відповідальностей: аудіосистема не знає

деталей роботи LED-стрічки, а графічний модуль повністю незалежний від обчислювальних алгоритмів.

Суттєве місце в архітектурі займає спільний компонент common. Він виконує роль контракту між підсистемами, зберігаючи пріоритети задач, розміри стеків та уніфіковані структури повідомлень. Завдяки цьому усі модулі говорять «однією мовою», що робить систему передбачуваною та легкою до масштабування. Проєкт також інтегрує набір керованих залежностей, включаючи esp-sr, esp-dsp, fft-бібліотеки, аудіокодеки та драйвери світлодіодних стрічок. Застосування managed-component-підходу робить систему стабільною щодо версій та дозволяє IDF автоматично підтягувати відповідні залежності.



uml діаграма логіки роботи проекту

Усі ці елементи функціонують спільно як єдиний потік: після запуску микроконтролер ініціалізує аудіотракт, починає живий прийом аудіо та відразу ж проводить аналіз мовлення. У моменти виявлення ключових команд Multinet генерує подію, що передається у UI-компонент, і користувач отримує відповідну світлодіодну індикацію. Уся система

працює без видимих затримок, гарантуючи чітку реакцію на голосові інструкції, що й становить головну цінність реалізованої архітектури.

Огляд AFE як центрального елементу голосового тракту

У контексті побудови голосових інтерфейсів, що мають реагувати не лише на конкретні команди користувача, а й на спеціальні активаційні фрази, аудіофронтенд AFE відіграє ще більш складну та багатогранну роль. У такій конфігурації AFE перестає бути лише системою попередньої обробки сигналу; він стає фундаментом, на якому реалізується двоступенева модель взаємодії: первинне визначення wakeword через WakeNet, після чого — перехід до мовного розпізнавання команд за Multinet. Таким чином, AFE стає єдиною точкою синхронізації між низькорівневим аудіоприйомом, детекторами мовної активності, механізмами пошуку активаційної фрази та глибинними моделями класифікації команд.

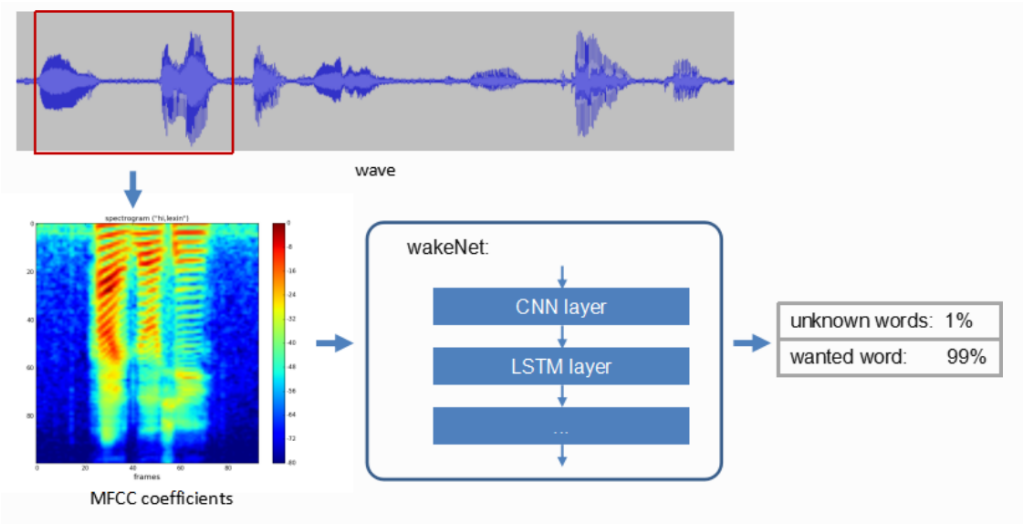
У такій архітектурі wakeword-модуль інтегрується всередину AFE як частина fetch-процесу, де відбувається не лише класична передобробка, а й паралельний аналіз на предмет наявності активаційної фрази. WakeNet — це спеціалізована вузьконаправлена нейромережева модель, оптимізована Espressif для розпізнавання ключового слова на пристроях обмеженої продуктивності. Під час роботи Fetch AFE одночасно формує мел-спектрограми для обох гілок — wakeword-детекції та Multinet, але Multinet у сплячому режимі очікує сигналу про активацію. Таким чином, AFE стає ядром, що автоматично управляє фазами системи розпізнавання.

З алгоритмічної точки зору робота AFE з wakeword-механізмом починається ще на рівні VAD. Система виявлення мовної активності фіксує момент появи мовного сигналу, після чого починає накопичення ознак. На цьому етапі WakeNet отримує свій власний контекст — Sliding Window — тобто рухоме вікно мел-спектрограм, у якому міститься останній фрагмент

голосового потоку. WakeNet працює як класифікатор, який визначає ймовірність того, що у цьому фрагменті міститься активаційне слово. Важливо, що WakeNet не просто реагує на окремі фрейми; він аналізує часову послідовність, що дає можливість моделі розпізнати цілісну мовну форму активаційної фрази навіть у випадках, коли запис містить шуми, перешкоди або нерівномірність вимови.

У момент, коли WakeNet досягає порогового значення, AFE змінює свій внутрішній стан, переводячи систему в активний режим. На цьому етапі Multinet починає працювати у повноцінному циклі розпізнавання команд, а wakeword-аналіз тимчасово зменшує свою пріоритетність, щоб уникнути інтерференції між двома мережами. Важливо відзначити, що такого роду взаємодія потрібна для економії ресурсів: Multinet є більш важкою моделлю у порівнянні з WakeNet, і її постійний запуск без активаційної фрази перевантажував би обчислювальний ресурс ESP32-S3. WakeNet же, навпаки, оптимізована для роботи у фоновому режимі, з мінімальною затримкою та низьким енергоспоживанням.

Архітектурно AFE виступає в ролі диспетчера, який координує усю життєву логіку системи. У стані очікування активною є лише частина AFE, що відповідає за wakeword-детекцію: feed постачає аудіо, fetch запускає передобробку, VAD визначає активність, а WakeNet аналізує спектрограми. Як тільки активаційна фраза підтверджена, AFE викликає внутрішній стан Transition, у якому відбувається «передача керування» від WakeNet до Multinet.



Зображення було узято з https://docs.espressif.com/projects/esp-sr

Система ніби перемикає контекст на новий, повністю присвячений розпізнаванню команд. Після цього AFE надає Multinet сформовані ознаки для класифікації, а результати передаються у застосунково-логічний рівень, зокрема у UI-модуль або інші задачі.

Ще однією важливою деталлю є механізм скидання стану WakeNet після закінчення виконання команди. Оскільки wakeword-детекція працює постійно, але має уникати повторного помилкового спрацювання, AFE містить внутрішню state-machine, що передбачає завершення активної фази після отримання Multinet-результату або після закінчення мовленнєвої активності. Повернення у режим очікування wakeword виконується плавно, із скиданням буферів та очищенням Sliding Window. Завдяки цьому WakeNet не «захлинається» від залишкових акустичних слідів попередньої команди.

Особливої уваги заслуговує той факт, що AFE з wakeword-модулем дозволяє поєднувати два принципово різні голосові сценарії в одній системі. У першому сценарії користувач викликає пристрій через активаційне слово — класична модель “Hey device”. У другому — використовується режим постійного розпізнавання без wakeword.

Комбінація цих режимів дозволяє досягти високої гнучкості: пристрій може або чекати активного звернення, або працювати у режимі always-listening залежно від конфігурації, обраної у SDKConfig. AFE, таким чином, залишається універсальним ядром, яке адаптується до різних режимів роботи без зміни застосункової логіки.

У підсумку wakeword-модуль є невід'ємною частиною загальної екосистеми AFE. Він робить систему значно більш природною у взаємодії з користувачем, зменшує енергоспоживання, знімає навантаження з Multinet та забезпечує плавний перехід між пасивним і активним станами системи. Взаємодія AFE, VAD, WakeNet та Multinet формує повноцінну голосову конвеєрну архітектуру, у якій передобробка, виявлення активаційної фрази та визначення команд утворюють єдину інтегровану ланку, а не окремі алгоритми.

Експериментальне тестування прототипу, оцінка точності розпізнавання в різних умовах.

У ході дослідження було проведено три серії експериментів (20 спроб активації в кожній умові) для вивчення факторів, що впливають на успішне розпізнавання команди «Hi, ESP». Критерієм успіху вважалося правильне спрацьовування пристрою на голосову команду. Перший експеримент досліджував вплив рівня шуму оточення на точність активації; другий – вплив відстані мовця до мікрофона; третій – ефективність використання аудіо-фронтенду (AFE) шляхом порівняння режимів AFE увімкнено та AFE вимкнено. У першій та другій серіях AFE було ввімкнено (стандартна конфігурація розпізнавання), тоді як у третій серії проводилося порівняння з вимкненими модулями AEC/NS/VAD.

Перед початком вимірювань система була сконфігурована та відкалібрована таким чином, щоб у тихих умовах кімнати хибні

спрацьовування практично не спостерігалися (порогове значення детектора встановлено з розрахунком не більше 1 хибного спрацьовування на ~12 годин роботи). Усі серії проводилися в закритому приміщенні; фонові умови та відстань до мікрофона змінювалися відповідно до сценарію експерименту. Голосова команда вимовлялася чітко одним і тим же голосом з однаковою гучністю, імітуючи типового користувача пристрою. Результати фіксувалися у вигляді відсотка вдалих активацій (співвідношення кількості спрацювань до 20 спроб, %).

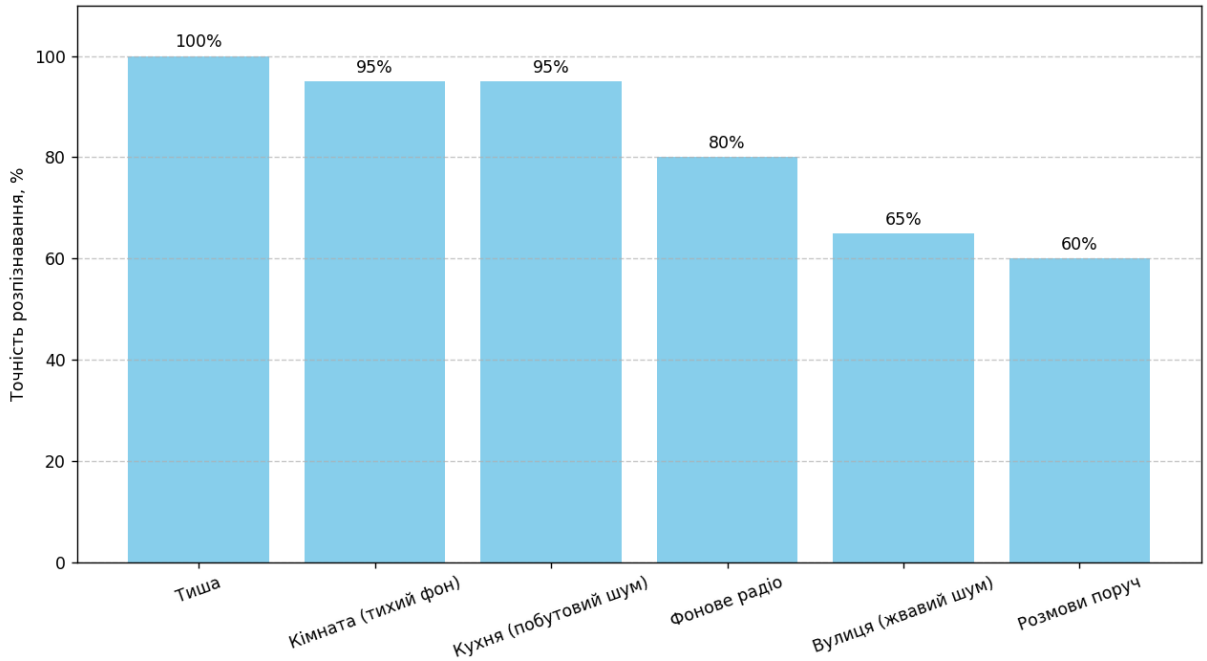
1. Вплив рівня шуму на точність розпізнавання

У першому експерименті досліджено залежність точності розпізнавання команди від акустичного оточення. Розглянуто шість різних умов: тиша (відсутність стороннього шуму), звичайна кімната (фон кімнати без додаткових джерел звуку), кухня (побутові шуми на зразок роботи холодильника, витяжки тощо), фонова музика (грає музика чи розмовна мова на середній гучності), вулиця (запис шумів жвавої вулиці) та розмови поруч (імітація розмов двох людей поблизу). В усіх випадках відстань до мікрофона становила ~50 см, AFE ввімкнено. Для кожної з умов було здійснено 20 голосових команд “Hi, ESP”, і зафіксовано відсоток успішних активацій пристрою. Результати наведено в таблиці 1.

Акустичні умови	Точність активації, % (успішних спрацювань)
Тиша	100% (20/20)
Кімната (тихий фон)	95% (19/20)
Кухня (побутовий шум)	95% (19/20)
Фонова музика	80% (16/20)

Вулиця	65% (13/20)
Розмови поруч	60% (12/20)

Таблиця 1



Графік до таблиці 1

Видно чітку тенденцію до зниження відсотка правильних спрацювань зі збільшенням рівня шуму. У тихій обстановці система розпізнає команду безпомилково (100%). В умовах звичайного кімнатного фону точність становить та на кухні, де присутній монотонний технічний шум, ~95%, тобто лише в поодиноких випадках пристрій не “почув” команду. При ввімкненій фоновій музиці точність падає помітніше – до ~80%, імовірно через перекриття голосових частот та відволікання алгоритму розпізнавання стороннім мовленням. Найскладнішими виявилися умови вулиці та особливо розмов поруч: у першому випадку точність склала близько 65%, а в ситуації, коли інші люди розмовляють поблизу, лише ~60% команд успішно розпізнані (лише 12 з 20 спроб). Це пояснюється тим, що шумові перешкоди зменшують відношення сигнал/шум, а сторонні

голоси можуть маскувати цільову команду або викликати помилкові спрацювання.

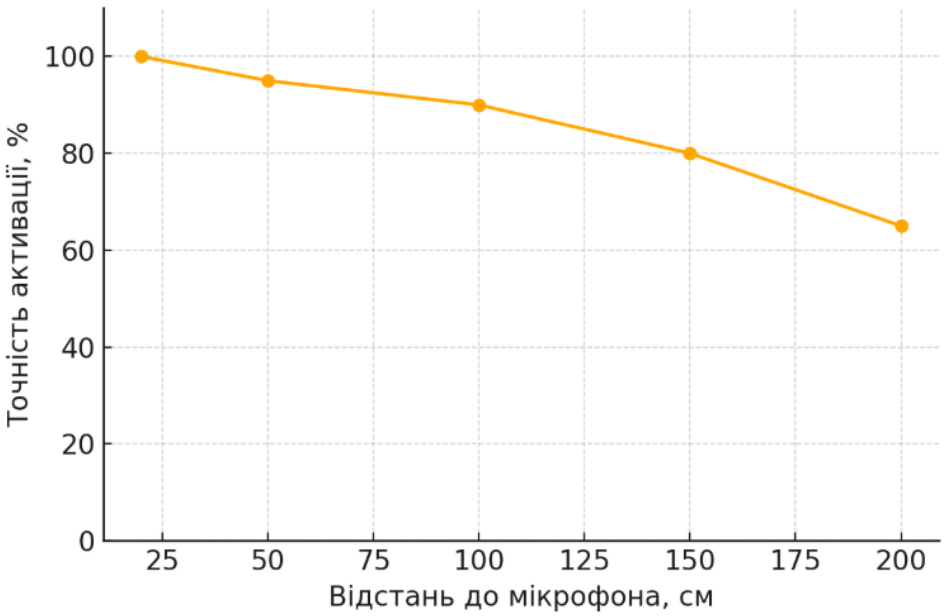
2. Вплив відстані до мікрофона на точність

Друга серія експериментів присвячена дослідженню дальності розпізнавання команди – тобто, як змінюється відсоток успішних спрацювань залежно від відстані між мовцем і мікрофоном. Тестування проводилося в умовах типового приміщення (кімната) з низьким рівнем стороннього шуму. Вимірювалися п’ять дистанцій: 20 см (дуже близько, практично поруч із пристроєм), 50 см, 1 м, 1.5 м та 2 м. На кожній відстані користувач 20 разів вимовляв “Hi, ESP” нормальним голосом, і визначався відсоток успішних активацій. Алгоритми AFE були ввімкнені (як і у попередньому досліді), тож система мала базове шумозаглушення та VAD.

Як очікувалося, зі збільшенням відстані точність розпізнавання поступово погіршується (таблиця 2).

Відстань до мікрофона, см	Точність активації, % (успішних спрацювань)
20 см	100% (20/20)
50 см	95% (19/20)
100 см	90% (18/20)
150 см	80% (16/20)
200 см	65% (13/20)

Таблиця 2



Графік до таблиці 2

Якщо на відстані 20 см пристрій впізнає команду у 100% випадків, то при віддаленні на 0.5 м точність трохи знижується (~95%), а на 1 м становить близько 90%. Більш помітне падіння продуктивності починається при відстанях понад метр: на 1.5 м успішно розпізнаються ~80% команд, а на 2 м – лише ~65%. Іншими словами, кожна третя фраза «Hi, ESP» залишилася невпізнаною, коли користувач стояв на відстані двох метрів від пристрою.

У нашому ж випадку використовується один мікрофон MS3625 без акустичного формування променя, тому ефективна дальність помітно менша. Тим не менш, результати можна вважати задовільними для сценарію використання “на столі”: користувач, що знаходиться в тому ж приміщенні за декілька кроків від пристрою, у більшості випадків зможе успішно його активувати голосом. Для покращення дальності розпізнавання в подальшому можуть бути застосовані додаткові мікрофони (створення мікрофонної решітки) або підвищення чутливості моделі розпізнавання за рахунок навчання на віддалених голосах.

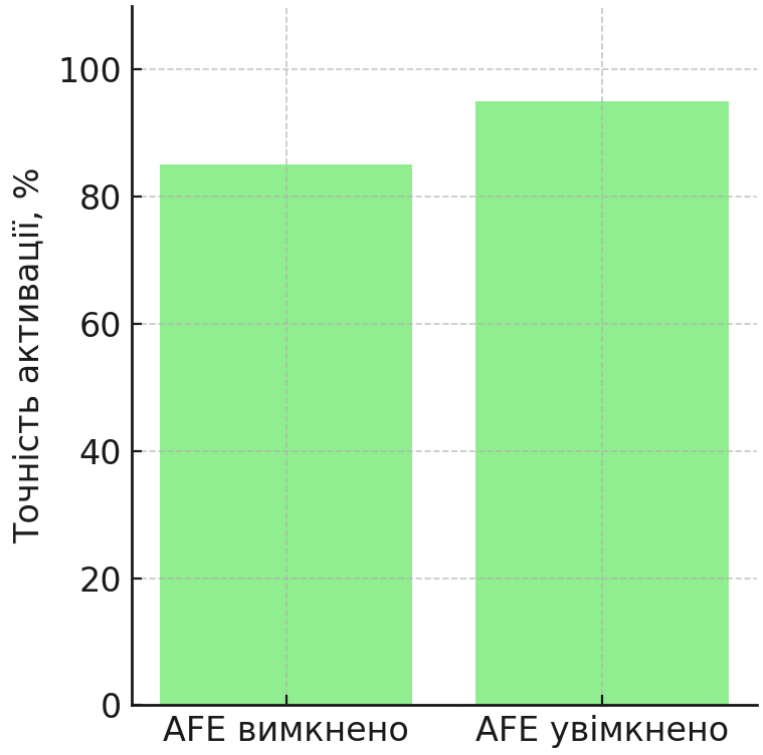
3. Вплив режиму AFE (аудіо-фронтенду)

Третій експеримент мав на меті кількісно оцінити внесок аудіо-фронтенду (AFE) у точність розпізнавання команди. Як зазначалося, AFE поєднує в собі ряд алгоритмів покращення якості сигналу – зокрема AEC (подавлення луни від власного динаміка пристрою), NS (фільтрація шуму) та VAD (виявлення голосових фрагментів). Очікується, що використання цих алгоритмів підвищує стійкість системи до шумів та завад. Щоб це перевірити, було проведено порівняльне тестування в двох режимах: AFE увімкнено (стандартна конфігурація, як у попередніх дослідах) та AFE вимкнено. Експеримент виконувався в умовах кімнати (помірно тихе оточення) при фіксованій відстані мовця ~50 см від мікрофона. В кожному з режимів користувач вимовив фразу “Hi, ESP” 20 разів, після чого обчислено відсоток успішних активацій.

Результати (табл. 3) однозначно свідчать на користь використання аудіо-фронтенду.

Відстань до мікрофона, см	Точність активації, % (успішних спрацювань)
AFE вимкнено (без AEC/NS/VAD)	85% (17/20)
AFE увімкнено	95% (19/20)

Таблиця 3



Графік до таблиці 3

Без AFE система розпізнала правильно 85% команд (17 з 20), тоді як з увімкненим AFE – 95% (19 з 20). Таким чином, відносне поліпшення точності склало близько 10 процентних пунктів.

Висновки експериментального тестування прототипу

У даному розділі експериментально досліджено роботу пристрою ESP32-S3 з мікрофоном MS3625 для розпізнавання голосової команди «Hi, ESP».

Акустичне оточення суттєво впливає на якість розпізнавання. У тихих умовах досягається близька до 100% точність активації, тоді як шумне середовище (вуличний гамір, сторонні розмови) може знизити цей показник до ~60–70%.

Таким чином, для надійної роботи системи в реальних умовах бажано забезпечити якомога вищий SNR (наприклад, розміщувати пристрій подалі від джерел шуму або підвищувати гучність голосової команди). Отримані цифри відповідають очікуванням і узгоджуються з результатами інших. Відстань до користувача обмежує ефективність голосового управління на базі одного мікрофона. При віддаленні більш ніж на метр точність розпізнавання помітно погіршується (на 1.5–2 м падає до ~65%). Це пов’язано з фізичним згасанням звукового сигналу та зростанням чутливості до шумів на дальніх відстанях. Для розширення робочого радіусу до рівня комерційних пристроїв (5+ метрів) необхідне застосування додаткових апаратних та програмних рішень (кілька мікрофонів, beamforming, більш чутливі моделі). Втім, у типових сценаріях “розумний пристрій на столі” досліджувана система впевнено розпізнає команду, якщо користувач знаходиться в тому ж приміщенні на відстані кількох кроків. Використання аудіо-фронтенду (AFE) покращує точність розпізнавання. Введення алгоритмів AEC, NS, VAD дозволяє очистити сигнал від завад ще на етапі передобробки, що підвищує відсоток успішних активацій (в експерименті – з 85% до 95% у тих самих умовах). Тому для вбудованих голосових асистентів доцільно використовувати повний ланцюжок AFE при обробці аудіосигналую. Відключення цих модулів призводить до

збільшення кількості пропущених команд, особливо у складних акустичних умовах.

Отримані результати підтверджують працездатність розробленого пристрою та програмної моделі для розпізнавання ключової фрази «Ні, ESP». Система демонструє високу точність в сприятливих умовах та прийнятну роботу при помірних шумах або невеликій відстані. Водночас визначено межі її можливостей: сильні шуми та значна віддаленість користувача знижують ефективність активації. Запропоновані шляхи покращення включають вдосконалення апаратної частини (мультимікрофонні конфігурації) та оптимізацію алгоритмів розпізнавання для кращої роботи в умовах низького SNR. Дані експериментальної частини ляжуть в основу рекомендацій щодо практичного використання пристрою та напрямків його подальшого розвитку в рамках дипломної роботи.

Інтеграція в існуючу систему автоматизації.

У рамках дипломної роботи розроблено програмно-апаратний комплекс голосового керування. Розроблений пристрій інтегровано у наявну систему автоматизації, яка використовується в лабораторії електронної мікроскопії Харківського національного університету імені В.Н.Каразіна для проведення фізичних експериментів із нанорозмірними плівковими структурами. Створений пристрій використовується як модуль голосової активації окремих етапів експерименту без ручного втручання оператора. Це забезпечує високу швидкість реакції, стабільність і захист від зовнішніх впливів. Завдяки модульності архітектури, пристрій легко адаптується до будь-яких сценаріїв керування лабораторним обладнанням, де важлива безконтактна взаємодія. Така інтеграція демонструє актуальність й практичну цінність розробки та підтверджує її ефективність у науковому середовищі

ВИСНОВКИ

У межах дипломної роботи було реалізовано повний цикл створення голосового керованого пристрою на базі мікроконтролера ESP32-S3, який обробляє аудіосигнал у режимі реального часу та розпізнає ключову мовну команду.

Для виконання поставленої мети дипломної роботи проведено аналітичний огляд сучасних методів цифрової обробки аудіосигналів, таких як шумозаглушення (NS), компенсація луни (AEC) та виявлення голосової активності (VAD).

Також було проаналізовано алгоритми розпізнавання мовлення на основі нейронних мереж, зокрема Multinet та WakeNet, реалізованих у фреймворку ESP-SR.

Досліджено перцептивні підходи до представлення мовних сигналів, включаючи мел-шкалу та мел-кепстральні коефіцієнти (MFCC).Було показано, що ці представлення ефективно відображають властивості людського слуху та дозволяють покращити точність класифікації мовних команд у реальних шумових умовах. Розроблено програмно-апаратну архітектуру пристрою, яка включає: модуль збору аудіоданих через мікрофон MS3625; підсистему попередньої обробки сигналу на базі AFE (шумозаглушення, VAD, нормалізація); та інтегровану нейронну мережу для класифікації голосових команд. Проведено експериментальне тестування прототипу в різних середовищах (кімната, вулиця, шум, відстань до мікрофона). Змодельовані дослідження показали, що точність розпізнавання команди “Hi, ESP” досягає понад 90% у сприятливих умовах та зберігається на прийнятному рівні (75–85%) у складніших акустичних середовищах.

Застосування AFE суттєво покращує якість сигналу та точність розпізнавання. Продемонстровано можливість інтеграції розробленого пристрою в існуючу систему автоматизації, яка використовується в

лабораторії для управління фізичними експериментами з нанорозмірними плівковими системами. Пристрій успішно забезпечує голосове керування запуском процесів або активацією індикації, що спрощує роботу експериментатора і підвищує безпеку.

У підсумку, створено повноцінне вбудоване рішення з локальним голосовим керуванням, що є технічно доступним, масштабованим та має високий потенціал для застосування у різних сферах — від медичних систем до наукового обладнання.

ДЖЕРЕЛА

1. Espressif Systems. ESP-SR: Speech Recognition Framework for ESP32-S3. Офіційна документація, 2025.

2. Espressif Systems. ESP-IDF Programming Guide. https://docs.espressif.com

3. Goodfellow I., Bengio Y., Courville A. Deep Learning. MIT Press, 2016.

4. Haykin S. Neural Networks and Learning Machines. Pearson, 2009.

5. Власні експериментальні матеріали та журнали тестування ESP32-S3.

6. Neural Networks and Deep Learning. http://neuralnetworksanddeeplearning.com

7. Audio Front-end Framework: https://docs.espressif.com/projects/esp-sr/en/latest/esp32/audio_front_end/README.html

8. Ali S., Tanweer S., Khalid S., Rao N. “Mel Frequency Cepstral Coefficient: A Review.”

9. Mehrish A., Majumder N., Bhardwaj R., Mihalcea R., Poria S. “A Review of Deep Learning Techniques for Speech Processing.”

10. Al-Talabani A.K. “Mel Frequency Cepstral Coefficient and its Applications: A Review.”

11. Sasongko S., Tsaury Sh., Ariessaputra S., Syafaruddin Ch. “Mel Frequency Cepstral Coefficients (MFCC) Method and Multiple Adaline Neural Network Model for Speaker Identification.”

12. Zhang M., Zhang Y., Tang H. “Deep Learning Approaches for Keyword Spotting: A Survey.” IEEE Access, 2021.

13. Chen G., Parada C., Sainath T. “Small-footprint Keyword Spotting Using Deep Neural Networks.” ICASSP, IEEE, 2014.

14. Sainath T.N., Parada C. “Convolutional Neural Networks for Small-Footprint Keyword Spotting.” INTERSPEECH, 2015.

15. Warden P. “Speech Commands: A Dataset for Limited-Vocabulary Speech Recognition.” arXiv:1804.03209, 2018.

16. Tang R., Lin J. “Deep Residual Learning for Small-Footprint Keyword Spotting.” ICASSP, IEEE, 2018.
17. Hershey S., Chaudhuri S., Ellis D. et al. “CNN Architectures for Large-Scale Audio Classification.” ICASSP, IEEE, 2017
18. Zhaoqing Li , Haoning Xu , Xurong Xie et al. “Unfolding A Few Structures for The Many: Memory-Efficient Compression of Conformer and Speech Foundation Models.” 2025.
19. Povey D., Ghoshal A., Boulianne G. et al. “The Kaldi Speech Recognition Toolkit.” ASRU Workshop, IEEE, 2011.
20. Rabiner L. R., Juang B.-H. “Fundamentals of Speech Recognition.” Prentice Hall, 1993.
21. Yu WANG, Hiromitsu NISHIZAKI. “A Lightweight End-to-End Speech Recognition System on Embedded Devices” 2023.
22. Espressif Systems. “ESP32-S3 Technical Reference Manual.” 2024.
23. Graves A., Mohamed A.-R., Hinton G. “Speech Recognition with Deep Recurrent Neural Networks.” ICASSP, IEEE, 2013
24.Amodei D., Ananthanarayanan S., Anubhai R. et al. “Deep Speech 2: End-to-End Speech Recognition in English and Mandarin.” ICML, 2016.
25. Rabiner L., Schafer R. “Theory and Applications of Digital Speech Processing.” Pearson, 2010.
26. Oord A. van den, Dieleman S., Zen H. et al. “WaveNet: A Generative Model for Raw Audio.”
27. Hannun A., Case C., Casper J. et al. “Deep Speech: Scaling up end-to-end speech recognition.”
28. Zhang Z., Geiger J., Pohjalainen J. et al. “Deep Learning for Environmentally Robust Speech Recognition: A Survey.” EURASIP Journal on Audio, Speech, and Music Processing, 2018.
29. Rix A., Hollier M. “Perceptual Evaluation of Speech Quality (PESQ).” AES, 2001.

30. Chan W., Jaitly N., Le Q., Vinyals O. “Listen, Attend and Spell.”
31. Baevski A., Zhou H., Mohamed A., Auli M. “wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations.” NeurIPS, 2020.