Київський національний університет імені Тараса Шевченко
Факультет комп'ютерних наук та кібернетики
Кафедра обчислювальної математики

ВИПУСКНА КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА

на тему:

Аналіз характеристик мультимодальних керованих моделей для стиснення відео

студента 4 курсу
Чомко Василя Дмитровича

Науковий керівники:
Асистент кафедри ОМ
Денисов Сергій Вікторович

Professor in the Department of Systems Design Engineering,
University of Waterloo
Paul Fieguth

Роботу розглянуто й допущено до захисту на засіданні кафеди обчислювальної математики *5 травня 2023 р., протокол № 7*

Завідувач кафедри ОМ Ляшко С.І.

Київ-2023

АНОТАЦІЯ

Мета цієї роботи полягає в визначенні оптимальної комбінації візуальних та аудіо характеристик для процесу стиснення відео, що базується на обробці мультимодальних ознак. Дослідження фокусується на відео матеріалах знятих на екшн-камери в польових умовах без професійного обладнання або подальшої постобробки. В порівнянні з унімодальним підсумовуванням відео (unimodal video summarization), дана сфера є недостатньо дослідженою. Одним з основних питань що виникають в даній області — є складність обробки великої кількості різноманітних ознак. В нашій роботі, ми, використовуючи дві моделі машинного навчання, Balanced Random Decision Forest та метод k-найближчих сусідів (k-nearest neighbor method, KNN), намагаємося визначити список найбільших важливих об'єднаних характеристик, що дозволить зменшити кількість даних, необхідних для обробки та дозволить зберегти рівень точності розглянутих моделей. Опираючись на результати, отримані в цьому дослідженні, ми відповімо на ключове питання: "Чому саме ці ознаки є важливими для стиснення відео?".

Ключові слова: важливість ознак, аналіз візуальних ознак, аналіз аудіо ознак, відео стиснення, мультимодальний аналіз, машинне навчання, збалансований ліс випадкових рішень, k-найближчі сусіди

ABSTRACT

Analysis of features of multimodal supervised models for video summarization. — Bach of Applied Science Degree Thesis by Speciality 113 «Applied Mathematics». — Taras Shevchenko National University of Kyiv of Ministry of Education and Science of Ukraine, Kyiv, 2023.

The aim of this study is to identify the optimal combination of visual and audio features for a video summarization process based on multimodal feature

processing. The study focuses on video materials captured by an action camera in an open environment without professional equipment or post-processing. The field of multimodal video summarization is relatively underexplored compared to unimodal video summarization, presenting challenges such as processing a large number of different features. In our research, we employ two machine learning models, Balanced Random Decision Forest and the k-nearest neighbour method (KNN), to identify the most significant combined features. This approach aims to decrease the volume of data needed for processing while preserving the accuracy of the models in question. The findings of this study seek to address the query: "What features are important for video summarization process?".

Key words: feature importance, visual feature analysis, audio feature analysis, video compression, multimodal analysis, machine learning, balanced random decision forest, k-nearest neighbors

ЗМІСТ

ВСТУП

За останні роки, значно збільшилася кількість відео, знятих за допомогою не професійної техніки (камери на телефоні, GoPro камери, камери в дронах і т.д.), кількість відзнятого подібним чином відеоматеріалу зростає експоненційно. Постає проблема обробки та аналізу такого об'єму даних. Переважна більшість методів обробки спирається лише на один канал інформації (аудіо/відео/текстовий). І основними причинами важкості аналізу є майже повна відсутність анотованих наборів даних та складність пропрацювання великої кількості різноманітних ознак. Вирішення останньої причини, ми намагаємося дослідити в даній роботі.

Наша робота полягає в глобальній оцінці важливості візуальних та аудіо ознак для задачі узагальнення відео контенту та визначення оптимальної комбінації найвагоміших об'єднаних характеристик, що дозволить зменшити обсяг даних, які потребують обробки, але при цьому зберегти потрібний рівень якості та точності моделей.

Предметом дослідження є відео матеріали зняті з екшн-камер в реальних умовах без професійного обладнання або додаткової постпродукції. Ми оброблюємо такі аспекти, як яскравість, контрастність, відсоток виявлених на зображенні облич, кількість різних побутових предметів на зображенні, міра невизначеності спектра звукового сигналу, міра розподілу частот в спектрі звукового сигналу та багато інших факторів.

В якості моделей для оцінки використовуємо керовану ансамблеву модель Balanced Random Decision Forest та керовану непараметричну модель k-найближчих сусідів (k-nearest neighbor method, KNN).

Результати нашого дослідження мають потенційне застосування в різних сферах. Вони можуть бути корисними для виділення ключових мо-

ментів з тривалих відеозаписів. Наприклад, нагрудні камери, що використовуються персоналом служби безпеки або правоохоронцями, створюють великий обсяг відеоматеріалів. Застосування методу узагальнення відео дозволяє ефективно виявляти критичні події та підозрілу активність.

Також ця технологія може бути корисною для журналістів та громадських репортерів, оснащених натільними камерами або навіть звичайними телефонами. Вони можуть записувати інциденти на місці подій, які потім можуть бути цікавими для новинних репортажів. Застосування методу узагальнення відео допомагає швидко відібрати та підкреслити найважливіші кадри, що сприяє більш ефективній розповіді та поширенню інформації.

СТРУКТУРА РОБОТИ

У 1 розділі ми надаємо опис основної проблеми, що розглядається у нашому дослідженні, та фокусуємося на питаннях, які ми плануємо дослідити в рамках даної роботи. Крім того, ми проводимо огляд літератури, розглядаємо різні підхід робіт та порівнюємо різні перспективи, з яких можна досліджувати питання стиснення відео та виділення ключових ознак.

У 2 розділі ми детально розповідаємо про теоретичний контекст предметної області, що стосується теми стиснення відео, виділення ознак, вибору ознак, дослідження методів кореляції. Також ми висвітлюємо теоретичний контекст об'єднання відео та аудіо ознак для використання на наступних етапах нашої роботи. Після цього ми переходимо до ключової теми нашого дослідження — визначення важливості ознак базуючись на їх оцінці. Ми наводимо теоретичний опис та математичне формулювання для двох моделей, які ми використовуємо: керовану ансамбльову модель Balanced Random Decision Forest та керовану непараметричну модель k-Nearest Neighbor method (KNN). В кінці 2 розділу ми надаємо детальний опис метрик, за якими будуть проходити процеси оцінки результатів роботи розроблених моделей.

У 3 розділі наводиться детальний опис технічної реалізації нашого дослідження. Ми починаємо з опису набору даних, який ми використовуємо та описуємо процес його збору, анотації та агрегації. Наступним кроком, ми описуємо використане програмне забезпечення та інструменти, які були необхідних для розробки та впровадження нашої моделі для стиснення відео та виділення ключових ознак. Також в 3 розділі, пояснюється структура та архітектура розробленої системи, включаючи різні компоненти та їх взаємодію. Наступним кроком є розгляд алгоритмів та методів, які ми

використовуємо для стиснення відео та виділення ключових ознак. Ми детально пояснюємо принципи роботи кожного алгоритму, його математичні моделі та параметризацію. Важливою частиною цього розділу є опис алгоритмів, які використовуються для автоматичного визначення важливості ознак та вибору оптимальної підмножини ознак для подальшого використання.

У 4 розділі ми приводимо результати отриманих метрик наших моделей. Ми надаємо порівняння результатів та пропонуємо свої пояснення щодо причин, чому ми отримали саме такі результати, і які фактори на це вплинули.

У 5 розділі ми формулюємо наші результати та перспективи для майбутніх робіт, які можуть внести вагомий внесок у майбутній розвиток сфери стиснення відео та виділення ключових ознак.

РОЗДІЛ 1
АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1. Огляд літератури

Прагнучи зрозуміти складнощі узагальнення відео, ми розпочали всебічне дослідження наявної літератури. Цей огляд виявив декілька якісних досліджень на цю тему, проте було виявлено помітну прогалину: детальне вивчення оцінки окремих ознак та їхнього подальшого впливу на остаточні метрики оцінки моделей відсутнє.

Наше дослідження почалося з вивчення стиснення мультимодального відео з використанням керованого навчання. У дослідженні "Навчання з учителем для узагальнення відео за допомогою використання кількісних наборів ознак із паралельною увагою"(Supervised Video Summarization Via Multiple Feature Sets with Parallel Attention) [1] було запропоновано нову модель для стиснення відео, що використовує три окремі набори ознак: візуальний контент, рух і механізм уваги. Ця модель визначає важливість кадрів або сегментів у відео, що є одночасно важливим і складним завданням. Механізм уваги застосовується до інтеграції ознак руху зі статичним візуальним контентом, тим самим покращуючи прогностичні можливості моделі для оцінки важливості. Ефективність моделі було продемонстровано під час масштабних оцінок на двох наборах даних, SumMe і TVSum, причому результати показали покращення для набору даних SumMe і відповідність найсучаснішим стандартам для набору даних TVSum.

Ще одне варте уваги дослідження, "Мультимодальне стиснення створених користувачем відео"(Multimodal Summarization of User-Generated Videos) [2], в якому розглядається підхід до узагальнення створених користу-

вачем відео з використанням як звукових, так і візуальних ознак. Автори стверджують, що більшість сучасних методів узагальнення відео недооцінюють можливості звукових характеристик і призначені для роботи переважно з комерційними/професійними відео. Їхній підхід спрямований на створення динамічних відеорезюме, які складаються з найбільш "важливих"частин оригінального відео, розташованих так, щоб зберегти їхній часовий порядок. У статті представлено бінарний класифікатор, який навчається розпізнавати важливі частини відео, використовуючи контрольовані знання з аудіо та візуальних модальностей. Також автори цього дослідження розробили та виклали в відкритий доступ новий анотований набір даних, що містить відео з 14 категорій знятих на не професійне обладнання, відео були отримані з платформи YouTube та анотовані 22 людьми, для визначення цікавих моментів відео. Анотований набір даних запропонований в цьому досліджені має безплатну ліцензію та автори надають дозвіл на його використовувати для дослідницьких робіт в суміжних сферах.

Одна з проаналізованих робіт фокусувалася на вирішенні пробелми неефективного навчання моделі без вчителля, через нерівномірний розподіл оцінок важиловсті ознак для подальшого узагальненя та проблему навчання на довгих відеозаписах. В статті “Дискримінативне навчання ознак для узагальнення відео з навчанням без вчителя” (Discriminative Feature Learning for Unsupervised Video Summarization) [3] автори розглядають проблему стиснення відео з навчанням без вчителя, а саме автоматичного виділення ключових кадрів з вхідного відеоматеріалу. Вони вирішують дві проблеми, які виникають під час цього процесу. Перша проблема полягає у неефективному навчанні через рівномірний розподіл вихідних оцінок важливості для кожного кадру. Автори пропонують вирішити цю проблему за допомогою втрати дисперсії, що є простим, але ефективним терміном втрати регуляризації. Це дозволяє мережі передбачати вихідні оцінки з великою розбіжністю, що покращує продуктивність моделі. Друга проблема

полягала у труднощах навчання з довгими вхідними відеозаписами. Для її вирішення автори розробили двопотокову мережу CSNet, яка використовує локальний (chunk) і глобальний (stride) часовий погляд на ознаки відео.

Дослідження "Неконтрольоване узагальнення відео за допомогою уважних умовних генеративних змагальних мереж"(Unsupervised video summarization with attentive conditional generative adversarial networks) [14] представило новий метод неконтрольованого узагальнення відео з використанням генеративних змагальних мереж (Generative Adversarial Networks, GANs). Авторами використовується методика GAN, в якій генератор генерує зважені ознаки кадру і прогнозує оцінки важливості на рівні кадру, тоді як дискримінатор розрізняє зважені та необроблені ознаки кадру. Умовний селектор ознак спрямовує GAN-модель на фокусування на більш важливих часових регіонах. У статті представлено використання алгоритму множинної самостійної уваги на рівні кадру для узагальнення відео, вивчення довготривалих часових залежностей і подолання локальних обмежень рекурентних комірок, таких як LSTMs. Метод показує кращі результати, ніж інші методи навчання без вчителя, і навіть деякі методи навчання з вчителем на наборах даних SumMe і TVSum.

У статті "Оптимізоване визначення подій та узагальнення для створення підсумків крикету з використанням підходу з емперським пінгвіном"(Emperor Penguin optimized event recognition and summarization for cricket highlight generation) [15] запропоновано унікальний підхід до аналізу аудіо та відео ознак, не шляхом прямого злиття ознак, а за допомогою підходу, в якому на першому етапі аналізуються аудіо ознаки, після чого аналізуються відео ознаки та виділяються найважливіші відео фрагменти.

У статті "Мультимодальна релевантність та поєднання ознак для стиснення фільмів на основі слухової, зорової та текстової інформації"(Multimodal Saliency and Fusion for Movie Summarization Based on Aural, Visual, and Textual Attention) [16] запропоновано метод виявлення цікавих аудіовізу-

альних сегментів у відеопотоці. Використовуються моделі оцінки важливості аудіо, візуальних та текстових анотацій. Метод оцінює звукову важливість на основі багаточастотних модуляцій сигналу та вимірює візуальну важливість з урахуванням простору, часу, інтенсивності, кольору та орієнтації. Текстову важливість визначають за маркуванням елементів транскрипту в субтитрах. Стаття також розглядає різні схеми злиття фільмових даних з мультимодальними анотаціями й надає порівняльні результати для різних модальностей.

Нарешті, в статті "Модель Bag-of-Importance з використанням навчання ознак на основі локалізованого обмеженого кодування для узагальнення відео"(A bag-of-importance model with locality-constrained coding based feature learning for video summarization) [17] представлено новий підхід до узагальнення відео, з акцентом на важливості локальних особливостей. Автори запропонували модель Bag-of-Importance (BoI), яка визначає ключові кадри на основі важливості локальних ознак, що відрізняється від традиційного узагальнення на рівні глобальних ознак.

Отже, огляд літератури надав всебічне розуміння поточного стану досліджень у галузі стиснення відео, створивши підґрунтя для наших власних досліджень у цій сфері.

1.2. Опис проблеми

В сфері аналізу та стиснення відео існує кілька ключових задач, які потребують вирішення для подальшого розвитку цієї теми. Незважаючи на доступність "сирих"відеоматеріалів, виникає проблема відсутності вільно доступних мультимодальних анотованих наборів даних. Такі набори даних були б надзвичайно корисними для розробки передових моделей, здатних обробляти різноманітні типи відеоматеріалів та надавати прогностичні висновки щодо важливості окремих фрагментів. Варто відзначити, що ця

область є відносно новою, і на цей момент не було проведено достатньо досліджень, які б задали напрямок та визначили ключові підходи для подальшого розвитку в цій сфері. У контексті застосування навчання з вчителем для підсумовування відео з відкритим вихідним кодом, на час проведення нашого дослідження, у відкритому доступі існує лише обмежена кількість наукових робіт і в них не представленні аспекти, які ми будемо обговорювати далі (детальніше в пункті 1.2 Огляд літератури).

Ключовою проблемою, яку ми намагаємося дослідити, є технічна складність та вартість обчислення та аналізу всіх мультимодальних ознак відео. Для вирішення нашою задачею було розробка нових алгоритмів та проведення аналізу, який дозволить зменшити кількості важливих ознак, що впливають на якість прогнозування важливих моментів у відео, зберігаючи точність моделі. Для цього ми використали дві моделі: керовану ансамблеву модель Balanced Random Decision Forest та керовану непараметричну модель k-найближчих сусідів (k-nearest neighbor method, KNN). Ці моделі були обрані, оскільки вони демонструють одні з найкращих показників точності та F-1 міри, при цьому залишаючи можливість для подальшого пояснення результатів моделі. Важливо підкреслити, що хоча ми могли обрати іншу ансамблеву модель, наприклад, XGBoost, яка дає дуже схожі результати до керованої ансамблевої моделі Balanced Random Decision Forest, ми вирішили, що підхід з використанням керованої непараметричної моделі KNN дозволить нам розглянути питання з різних концептуальних сторін та матиме більше фундаментальне та широке розуміння отриманих значень важливості ознак. Також, ми вирішили не використовувати глибокі нейронні мережі, оскільки намагалися уникнути проблеми "чорної скриньки коли вихідні прогнози моделі не пояснюються та не можуть бути інтерпретовані, і зрозуміти, що саме впливає на отримання таких результатів.

Наступним етапом дослідження є вивчення вагомості окремих ознак зі

списку всіх мультимодальних ознак та їх впливу на точність та F-1 показник моделі. Ми досягли цього за допомогою методу перестановок та методу виключення ознак. Для отримання даних про важливість кожної окремої ознаки та розуміння, які з них можна прибрати, ми розрахували показник коваріації для кожної з них.

У кінцевій частині дослідження ми проводимо аналіз та надаємо обґрунтовані відповіді на наступні питання: причини значного впливу окремих ознак на фінальні результати моделі. Також, ми аналізуємо та відповідаємо на питання, чому при зменшенні деяких ознак, це не призводить до погіршення досягнутого попереднього рівня точності моделі.

РОЗДІЛ 2
ТЕОРЕТИЧНА ЧАСТИНА

Дослідження починається з огляду процесу та методів, що використовуються при стиснення відео (2.1), за яким слідує обговорення методів, що використовуються для вилучення релевантних ознак з відеоконтенту (2.2 Виділення ознак). Далі ми детально розглянемо процес розпізнавання шаблонів та відбору найбільш значущих ознак для стиснення відео "Розпізнавання шаблонів та вибір ознак"(2.3). У цьому підрозділі розглядається, як розпізнаються шаблонів серед обраних ознак (2.3.1), критерії та методи, що використовуються при відборі ознак (2.3.2), а також застосування методу Wrapper при відборі ознак (2.3.3). Також в цьому підрозділі проаналізовано кореляцію різних ознак (2.3.4), з детальним поясненням того, як коефіцієнт кореляції Спірмена використовується для розуміння взаємозв'язку між ознаками (2.3.4.1).

Потім дослідження переходить до процесу вилучення та об'єднання ознак (2.4 Мультимодальне вилучення та об'єднання ознак). Розглядаються специфіки аналізу ознак з відео (2.5) та аудіоконтенту (2.6 "Ознаки аудіо"), після чого обговорюється значимість різних ознак у процесі стиснення відео та можливості оцінювання ваги (2.7).

Розглянуто застосування машинного навчання в цьому дослідженні (2.8), з поясненням використаних методів керованого навчання (2.9 Навчання з вчителем). Детально обговорюється використання випадкового лісу (2.10 Decision Forest), включаючи метод дерева рішень (2.10.1 Decision Tree), часову складність алгоритму дерева рішень (2.10.2), можливі алгоритми, що використовуються для побудови дерев рішень (2.10.3), способи поєднання ознак у алгоритмі дерево рішень (2.10.4), використання ансамблевих мето-

дів (2.10.5) та використання лісу випадкових рішень (2.10.6).

Перший розділ завершується поясненням використаних методів навчання без вчителя (2.11), детальним обговоренням використання методу k-найближчих сусідів (KNN) у дослідженні (2.12), а також вивченням метрик, використаних для оцінки ефективності процесу узагальнення відео (2.13 Метрики оцінювання).

2.1. Стиснення відео

Стиснення відео (Video Summarization) - це процес скорочення та узагальнення відеоконтенту до компактного представлення, що включає найважливіші аспекти. Головною метою стиснення відео є забезпечення швидкого та ефективного перегляду великого обсягу відеоданих, при цьому зберігаючи інформативний огляд контенту.

У процесі стиснення відео використовуються різноманітні підходи та методи. Зокрема, підходи можна розділити на унімодальні та мультимодальні в залежності від того, яка модальність враховується під час стиснення.

Унімодальні підходи зосереджені на одній конкретній модальності, такі як візуальна, аудіо або текстова інформація. Розглянемо кілька унімодальних методів:

Екстрактивний метод, який полягає у виборі ключових кадрів або сегментів, що найкраще представляють контент. Ключові елементи вибираються на основі різних критеріїв, таких як візуальні зміни, рух об'єктів, аудіохарактеристики або текстова інформація. Шляхом виділення найважливіших та найрепрезентативніших моментів, екстрактивне стиснення надає компактну версію відео, яка зберігає його сутність.

Генеративних метод, який передбачає створення нових кадрів або сцен для формування підсумкового відео. Цей метод включає використання генеративних моделей, таких як автокодувальники (autoencoder) або генера-

тивні змагальні мережі (Generative adversarial networks, GAN), для створення компактних представлень, які зберігають основну інформацію відео. Генеративне стиснення відео має важливу перевагу: воно дозволяє створювати нові та компактні підсумки, що виходять за межі простого вибору та упорядкування існуючих кадрів.

Кожен з унімодальних підходів фокусується на використанні одного з типів модальності:

1. Візуальний підхід: Цей підхід базується на аналізі візуальної інформації відео. Він використовує різні техніки комп'ютерного зору, такі як визначення ключових кадрів на основі зміни контенту, розпізнавання об'єктів або аналіз текстурних атрибутів. Цей підхід дозволяє виділяти важливі кадри або сегменти на основі їхньої візуальної значущості.

2. Аудіо підхід: Аудіо підхід зосереджений на аналізі звукової інформації відео. Він використовує алгоритми обробки звуку для виявлення ключових аудіофрагментів, які вказують на важливі події або зміни відтворюваного звуку. Цей підхід є корисним у випадках, коли звук має велику вагу у відеоконтенті, наприклад, у випадку музичних виступів або голосового коментаря.

3. Текстовий підхід: Текстовий підхід використовує аналіз текстової інформації, пов'язаної з відео, такої як заголовки, описи або автоматично отримані підписи. Шляхом обробки текстових даних можна ідентифікувати ключові теми, інформативні ключові слова або фрази, що допомагають визначити важливість певних сегментів відео.

Також існують мультимодальні підходи, які враховують декілька модальностей відео, такі як візуальна, аудіо та текстова інформація. Шляхом комбінації даних з різних модальностей, таких як використання обробки зображень, модальностей, звуку та аналізу тексту, можна отримати більш повні та інформативні підсумки відео. Мультимодальні підходи забезпечують більш комплексний огляд відеоданих.

Серед мультимодальних підходів, можна виділити наступні:

1. Візуально-аудіо підхід: Цей підхід поєднує аналіз візуальної та аудіо інформації відео. Використовуючи техніки обробки зображень та звуку, він спробує виявити візуальні та звукові елементи, які найкраще відображають зміст відео.
2. Візуально-текстовий підхід: Цей підхід комбінує аналіз візуальної та текстової інформації. Використовуючи методи комп'ютерного зору та обробки тексту, він визначає візуальні атрибути та ключові слова, які найкраще репрезентують контент відео.
3. Візуально-аудіо-текстовий підхід: Цей підхід поєднує всі три модальності - візуальну, аудіо та текстову. Використовуючи комплексний аналіз всіх доступних даних, він забезпечує найбільш повну та інформативну версію підсумку відео.

Унімодальні підходи корисні для аналізу конкретних модальностей, а мультимодальні підходи дозволяють отримати більш комплексний та повний огляд відео. Кожен з цих підходів має свої переваги, обмеження та відповідає різним ситуаціям та завданням у стисненні відео. Оптимальний вибір підходу залежить від контексту відео, доступності даних та вимог застосування. Комбінація різних підходів може призвести до досягнення найкращих результатів у стисненні відео та створенні змістовних підсумків.

Для ефективного стиснення відео використовуються різні алгоритми та моделі, включаючи навчання з учителем (supervised learning), навчання без учителя (unsupervised learning) та навчання з обмеженою участю (weakly supervised learning). Кожен з цих методів має свої особливості та використовує різні підходи для визначення важливого контенту відео.

Методи навчання з учителем використовують анотовані тренувальні дані, де експерти вручну маркують ключові кадри або сегменти, щоб навчити моделі точно визначати важливий контент. Ці анотації слугують основою

для тренування моделі, яка навчається розпізнавати певні візуальні, аудіо або текстові ознаки, що вказують на важливість кадрів або сегментів. Після навчання модель може застосовуватись для автоматичного визначення ключових моментів в нових відеоданих.

Методи навчання без учителя натомість мають за мету виявити значний контент та структуру відео без попередніх знань або анотацій. Вони використовують різні підходи, такі як кластеризація, графові алгоритми або методи оптимізації, щоб визначити репрезентативні кадри або сегменти, які найкраще представляють вміст відео. За допомогою кластеризації відео розбивається на групи сегментів зі схожими візуальними або звуковими характеристиками, визначаючи ключові моменти. Графові алгоритми та методи оптимізації дозволяють моделі знайти оптимальний набір кадрів або сегментів, які найкраще відображають зміст відео.

Методи навчання з обмеженою участю знаходяться між навчанням з учителем та навчанням без учителя. Вони використовують часткові або неточні анотації, такі як мітки на рівні відео або взаємодію користувача, щоб керувати процесом стиснення. Наприклад, користувач може вказати певні моменти відео або позначити області інтересу, і модель використовує ці вказівки для визначення важливого контенту. Цей підхід дозволяє поєднати експертні знання та автоматичний аналіз для досягнення кращих результатів.

Використання різних методів навчання в стисненні відео дозволяє забезпечити більш точне та інформативне стиснення, зберігаючи сутність відеоконтенту. Оптимальний вибір методу залежить від доступних даних, характеристик відео та вимог застосування. Комбінація різних підходів та методів навчання може допомогти досягти кращих результатів в стисненні відео та створенні змістовних підсумків.

2.2. Виділення ознак

Виділення ознак (англ. feature extraction) є ключовим етапом у процесі обробки даних, який спрямований на видобування важливих ознак зі вхідних даних. Ознаки є абстрактними представленнями даних, які виражають специфічні характеристики або властивості, що можуть бути корисними для подальшого аналізу та використання у моделях машинного навчання.

Основна ідея виділяння ознак полягає в тому, щоб перетворити складні вхідні дані в простори ознак меншої розмірності, зберігаючи при цьому важливу інформацію. Це сприяє зменшенню обсягу даних та спрощенню подальшого аналізу. Один з головних принципів — це виділення репрезентативних ознак, які мають високу варіативність у межах даних та добре розрізняють різні класи або категорії. Існує безліч методів виділення ознак з вхідних даних.

Один із підходів — це використання математичних методів, таких як статистичні показники, фільтри або трансформації даних. Наприклад, для обробки зображень можна використовувати дескриптори форми, текстурні ознаки, градієнтні характеристики тощо. Для аудіо сигналів можна використовувати спектральні аналізатори, характеристики ритму або тембру. Текстові дані можна піддавати обробці, використовуючи методи тематичного моделювання, векторні представлення слів або семантичні аналізатори.

Інший підхід — це використання попередньо навчених моделей, зокрема нейронних мереж. Це дозволяє отримати вектори ознак, які мають високу абстрактність та здатні розпізнавати складні патерни у даних. Наприклад, з використанням глибокого навчання можна витягнути ознаки зображень за допомогою популярних архітектур, таких як згорткові нейронні мережі (convolutional neural network, CNN). Для обробки природної мови викори-

стовуються моделі, такі як рекурентні нейронні мережі (recurrent neural networks, RNN) або трансформер архітектури (Transformer-based).

2.3. Розпізнавання шаблонів та вибір ознак

2.3.1. Розпізнавання шаблонів.

Розпізнавання шаблонів є широко застосовуваною технологією у різних галузях, до прикладу в обробці зображень. У цій сфері існують певні виклики, такі як обробка великого обсягу зображень, які містять мільйони пікселів. Адже, використання традиційних методів, таких як вирахування власних значень коваріаційних матриць, для таких великих обсягів даних може бути обчислювально складним.

Окрім того, об'єкти, які потрібно класифікувати, наприклад, автомобілі, обличчя або тварини, можуть з'являтися у зображенні в різних місцях, з різними кутами обертання та освітленням. Для ефективного розпізнавання цих об'єктів необхідна велика кількість зразків, що охоплюють різноманітні варіації. Більшість пікселів у зображенні вважаються фоном і не є важливими для класифікації.

В контексті обробки зображень було розроблено багато методів та функцій, таких як визначення контурів, кутів, ліній/форм і масштабно-інваріантних ознак. Ці методи допомагають виділити ключові характеристики зображень і полегшують процес класифікації. Розуміння цих методів в області обробки зображень та комп'ютерного бачення є важливим для успішного використання технології розпізнавання шаблонів.

При обробці звукових сигналів також стикаємося з подібними викликами. Наприклад, якщо ми маємо запис мовлення тривалістю 50 секунд із частотою дискретизації 10 кГц, то ми отримуємо 500 000 вибірок даних. Як і у випадку обробки зображень, аналізувати 500 000 $\times$ 500 000 коваріаційних матриць не є практично можливим.

Для розв'язання цих проблем звукові сигнали зазвичай поділяються на невеликі сегменти, припускаючи, що кожен сегмент містить окремий звуковий елемент. Для отримання корисних характеристик використовується кепстральний аналіз.

Кепстральний аналіз використовує перетворення Фур'є, щоб перевести звуковий сигнал у спектральне представлення. Після цього застосовується зворотне перетворення Фур'є до логарифма абсолютного значення спектра. Таким чином, отримується вектор кепстральних коефіцієнтів, який може бути використаний для розпізнавання звуків та подальшого аналізу мовлення 2.1

$$\underline{x}_{\text{Cepstrum}} = \left|\mathcal{F}^{-1}\{\log(|\mathcal{F}(\underline{y})|)\}\right| \tag{2.1}$$

Відокремлення звукових сегментів та застосування кепстрального аналізу дозволяє отримати компактне представлення звукових характеристик, що використовується для розпізнавання мовлення та подальшого аналізу.

2.3.2. Вибір ознак.

У сфері машинного навчання та аналізу даних широко застосовується вибір ознак (feature selection) - процес обрання підмножини ознак з початкового набору даних. Вибір ознак має на меті зменшити розмірність простору ознак, зберігаючи при цьому важливу інформацію та знижуючи складність моделювання.

Для точного вираження ідеї вибору ознак, можна розглянути наступну формулу. Нехай X буде множиною вхідних ознак, Y - цільова змінна, а f(X) - цільова функція, яку ми намагаємося максимізувати або мінімізувати. Тоді, задача вибору ознак може бути сформульована наступним чином 2.2:

$$\text{maximize/minimize } J(X) \text{ subject to } X \subseteq \mathcal{P}(X) \tag{2.2}$$

де J(X) - критерій, що відображає якість набору ознак X, а $\mathcal{P}(X)$ - множина всіх підмножин множини X. Задача полягає в пошуку оптимальної підмножини ознак, що максимізує або мінімізує критерій J(X).

Однією з переваг вибору ознак є збереження інтерпретованості та одиниць вимірювання початкових ознак. Кожна обрана ознака відображає конкретний аспект досліджуваного явища і зберігає його значення у відповідних одиницях. Це робить отримані результати легкими для розуміння та інтерпретації.

Вагомою перевагою вибору ознак є зменшення кількості необхідних вимірювань. Якщо початковий набір даних містить велику кількість ознак, то за допомогою вибору певних ознак можна ефективно зменшити обчислювальну складність та ресурси, що вимагаються для обробки та аналізу даних.

У порівнянні з вилученням ознак (Feature extraction), яке базується на побудові нових ознак шляхом комбінування початкових, вибір ознак використовує лише початкові ознаки без створення нових. Це дозволяє зберегти первинну інформацію та властивості початкових даних.

Однак, вибір ознак також має свої обмеження. Це дискретний процес, де кожна ознака може бути використана або не використана, що може привести до втрати певної інформації. Крім того, оптимізація вибору ознак є складною задачею через комбінаторну природу пошуку оптимальної підмножини ознак.

Математично, вибір ознак можна представити як оптимізаційну задачу, де метою є максимізація критерію, що враховує важливість ознак та зменшення розмірності простору ознак. Цю задачу можна формалізувати наступним чином:

$$\max_S J(S)$$

де S - множина обраних ознак, $J(S)$ - критерій, який враховує як важливість обраних ознак, так і розмірність простору ознак. Оптимальна підмножина ознак S^* знаходиться шляхом пошуку розв'язання цієї задачі

В контексті даної проблеми також важливо мати на увазі наступні формули:

1. Коефіцієнт кореляції: Коефіцієнт кореляції використовується для визначення взаємозв'язку між ознаками і цільовою змінною. Це можна виразити формулою 2.3:

$$\rho\left(X_i, Y\right) = \frac{\operatorname{cov}\left(X_i, Y\right)}{\sigma_{X_i} \cdot \sigma_Y} \tag{2.3}$$

де X_i - і-та ознака, Y - цільова змінна, $\operatorname{cov}(X_i, Y)$ - коваріація між X_i і Y, а $\sigma_{X_i} \cdot \sigma_Y$ - стандартне відхилення відповідно для X_i і Y.

2. Інформаційний критерій: Інформаційний критерій використовується для визначення значущості ознаки з точки зору інформаційного виграшу або втрати. Це можна виразити формулою 2.4:

$$I\left(X_i\right) = \sum_{c \in C} p(c) \cdot \log\left(\frac{1}{p\left(c \mid X_i\right)}\right) \tag{2.4}$$

де X_i - і-та ознака, C - множина класів, $p(c)$ - апріорна ймовірність класу , а $p\left(c \mid X_i\right)$ ймовірність класу за умови X_i.

3. Дисперсія ознаки: Дисперсія ознаки відображає міру розкиду значень ознаки. Це можна виразити формулою 2.5:

$$\operatorname{Var}\left(X_i\right) = \frac{1}{n} \sum_{i=1}^{n} \left(X_i - \mu_{X_i}\right)^2 \tag{2.5}$$

Де X_i - і-та ознака, n — кількість спостережень, а μ_{X_i} - середнє значення ознаки X_i.

Ці формули дозволяють виміряти значущість ознак та їхній вплив на модель.

2.3.3. Метод Wrapper.

Метод Wrapper є підходом до вибору ознак, який оцінює продуктивність певної моделі машинного навчання шляхом розгляду підмножин ознак. На відміну від підходу фільтрації, який ранжує ознаки на основі їх індивідуальних характеристик, метод Wrapper використовує прогностичну модель для оцінки якості підмножин ознак. Його метою є вибір оптимального набору ознак, який максимізує продуктивність обраної моделі.

Математично метод Wrapper можна представити наступним чином. Розглянемо набір даних з n вибірок та m ознак, позначений як

$$X = (x_1, x_2, ..., x_m)$$

, де x_i представляє і-тий вектор ознак. Для знаходження найкращої підмножини ознак, S, яка максимізує продуктивність певної моделі машинного навчання, позначеної як M, задача полягає в розв'язання оптимізаційної задачі:

$$S^* = argmax_S score(M(X_S))$$

де $M(X_S)$ представляє показник продуктивності моделі M, навченої на під наборі ознак X_S, а $score(.)$ є функцією оцінки, яка оцінює продуктивність моделі. Змінна S^* представляє оптимальний під набір ознак, вибраний методом Wrapper.

Для пошуку оптимального під набору метод Wrapper використовує пошуковий алгоритм, такий як forward selection, backward elimination або exhaustive search. Ці алгоритми досліджують різні комбінації ознак та оцінюють їх продуктивність за допомогою крос-валідації або іншої техніки оцінювання. Пошуковий алгоритм ітеративно додає або видаляє ознаки з поточного під набору до тих пір, поки не буде знайдено найкращий під набір з найвищим показником продуктивності.

Метод Wrapper має перевагу у тому, що він враховує взаємодію та залежності між ознаками, що дозволяє захопити складніші взаємозв'язки

в даних. Однак він може бути обчислювально витратним, особливо коли простір ознак є великим. Крім того, вибір моделі машинного навчання в методі Wrapper може вплинути на кінцеві результати, оскільки різні моделі можуть мати різну чутливість до різних під наборів ознак.

2.3.4. Дослідження кореляції.

Ознаки бувають корельовані та не корельовані.

Кореляція між ознаками вказує на наявність статистичної залежності між ними. Це означає, що коливання значення однієї ознаки супроводжуються відповідними змінами в інших ознаках. У випадку сильної кореляції між двома ознаками, можна очікувати, що зміна значення однієї ознаки буде супроводжуватися аналогічною зміною в іншій ознаці.

Сильна кореляція між ознаками може впливати на якість моделей або алгоритмів машинного навчання. Наприклад, це може призвести до збільшення складності моделі або до спотворення результатів, оскільки деякі ознаки можуть надавати зайву або дубльовану інформацію.

Одним з можливих розв'язань проблеми кореляції ознак є їх виключення з моделі або алгоритму. При цьому можна залишити лише одну з корельованих ознак або обрати підмножину ознак, які мають найсильніший вплив на цільову змінну або які є незалежними одна від одної. Виключення деяких ознак може допомогти зменшити складність моделі й зберегти важливу інформацію.

Однак, перед виключенням корельованих ознак необхідно провести аналіз та оцінку впливу на якість моделі. Іноді кореляція може бути природною інформацією, а видалення таких ознак може призвести до втрати важливої інформації.

Іншим варіантом розв'язання проблеми кореляції ознак є використання методів перетворення ознак, таких як Метод головних компонент (Princi-

pal Component Analysis, PCA) або регуляризація. Ці методи дозволяють зменшити вплив кореляції шляхом створення нових некорельованих ознак.

2.3.4.1. Коефіцієнт кореляції Спірмена з ранговим порядком.

Коефіцієнт кореляції Спірмена з ранговим порядком - це статистична міра, яка використовується для вимірювання ступеня залежності між двома змінними, коли дані представлені у вигляді рангових порядків. Цей коефіцієнт базується на порівнянні рангових позицій двох змінних, замість самого числового значення.

У формулі коефіцієнта кореляції Спірмена $\rho_{\mathrm{R}(X),\mathrm{R}(Y)}$, ρ позначає звичайний коефіцієнт кореляції Пірсона, але використовується для рангових змінних $\mathrm{R}(X)$ та $\mathrm{R}(Y)$. Цей коефіцієнт відображає міру статистичної залежності між рангами двох змінних.

Коваріація рангових змінних $\mathrm{cov}(\mathrm{R}(X), \mathrm{R}(Y))$ в чисельнику вимірює ступінь спільної зміни між рангами $\mathrm{R}(X)$ та $\mathrm{R}(Y)$. Вона враховує, як одна змінна змінюється відносно іншої.

Стандартні відхилення рангових змінних $\sigma_{\mathrm{R}(X)}$ та $\sigma_{\mathrm{R}(Y)}$ в знаменнику вимірюють розкид значень рангів для кожної змінної окремо. Вони вказують на розмаїтість і варіабельність рангів.

Формула для обчислення коефіцієнта кореляції Спірмена r_s базується на цих компонентах і обчислюється як відношення коваріації рангових змінних до добутку їх стандартних відхилень:

$$r_s = \frac{\mathrm{cov}(\mathrm{R}(X), \mathrm{R}(Y))}{\sigma_{\mathrm{R}(X)}\sigma_{\mathrm{R}(Y)}}$$

Також існує альтернативна формула для обчислення r_s, яка може бути використана, якщо всі n рангів є різними цілими числами. Ця формула використовує різниці між рангами кожного спостереження $d_i = \mathrm{R}\left(X_i\right) -$

$\mathrm{R}(Y_i)$ і обчислює r_s як:

$$r_s = 1 - \frac{6 \sum d_i^2}{n\left(n^2 - 1\right)}$$

де n - кількість спостережень.

Коефіцієнт кореляції Спірмена є непараметричною мірою кореляції, що означає, що він не вимагає жодних припущень щодо розподілу даних. Це дозволяє використовувати його в широкому спектрі задач, особливо в тих випадках, коли дані не мають лінійної залежності або містять викиди.

2.4. Мультимодальне вилучення та об'єднання ознак

Мультимодальне вилучення ознак (Multi-modal feature extraction) є процесом витягування корисних ознак з різних типів даних або наборів даних. У контексті багатомодальних даних, це означає, що у нас є різні типи інформації, такі як зображення, звук, текст тощо, і ми хочемо виділити корисні ознаки з кожного типу даних, щоб отримати комплексне розуміння досліджуваного явища або задачі.

Процес мультимодального вилучення ознак може містити застосування різних алгоритмів та методів обробки для кожного типу даних. Наприклад, для зображень можуть бути використані алгоритми комп'ютерного зору для виявлення особливих точок або витягування текстурних ознак. Для звукових даних можуть бути використані алгоритми аналізу звуку для виділення акустичних характеристик. Для текстових даних можуть бути використані алгоритми обробки природної мови для витягування семантичних ознак.

Наведено кілька загальних формул, що можуть бути застосовані для процесу мультимодального вилучення ознак:

1. Зображення (Image): - Local Binary Patterns (LBP):

$$\mathrm{LBP}\left(x_c\right) = \sum_{n=0}^{N-1} s\left(g_n - g_c\right) 2^n$$

де x_c - центральний піксель, g_c - значення пікселя в центральній точці, g_n— значення пікселя на сусідніх точках, $s(\cdot)$ - функція порівняння. - Scale-Invariant Feature Transform (SIFT):

$$\text{SIFT}(x, y, \sigma) = \sum_{i=1}^{m} \sum_{j=1}^{m} \sum_{k=1}^{n} D(i, j, k) G(x - i, y - j, \sigma - k)$$

де x, y - координати пікселя, σ - масштаб, m, n - розміри фільтра, $D(\cdot)$ - дескриптор, $G(\cdot)$ - гаусіанове ядро.

2. Звук (Audio): - Mel-Frequency Cepstral Coefficients (MFCC):

$$\text{MFCC}(x(t)) = \text{IDCT} \left\{ \log \left[\sum_{k=1}^{N} |\text{DFT} x(t)|^2 \right] \right\}$$

де $x(t)$ - аудіо сигнал, $\text{DFT}(\cdot)$ - дискретне перетворення Фур'є, IDCT $(\cdot)$ - зворотне дискретне косинусне перетворення.

3. Текст (Text): - Term Frequency-Inverse Document Frequency (TF-IDF):

$$\text{TF} - \text{IDF}(t, d, D) = \text{tf}(t, d) \times \text{idf}(t, D)$$

де t - термін, d - документ, D - колекція документів, $\text{tf}(\cdot)$ - частота терміна в документі, $idf()$- інверсна частота документа.

Після отримання ознак з кожного типу даних, вони можуть бути поєднані в одну комплексну ознакову представлення, яке може бути використане для подальшого аналізу, класифікації або розпізнавання.

Об'єднання ознак (features fusion) є потужним підходом для поліпшення якості стиснення відео шляхом комбінування та об'єднання важливих ознак з різних джерел або модальностей. Цей підхід дозволяє врахувати різні аспекти та характеристики відео для отримання повнішої та репрезентативної інформації.

У контексті об'єднання ознак, існують два підходи — раннє об'єднання ознак (early feature fusion) та пізнє об'єднання ознак (late feature fusion).

Рання об'єднання ознак (early feature fusion) відбувається на початковому етапі обробки відео, коли різні модальності або джерела ознак з'єднуються в єдиний представлення. Цей підхід дозволяє використовувати об'єднані ознаки протягом усього процесу стиснення відео, що може призвести до зменшення обчислювальних витрат та покращення ефективності.

Математично, рання об'єднання ознак можна виразити за допомогою вагового об'єднання (weighted fusion) або множення ознак (feature multiplication). Вагове об'єднання передбачає призначення ваги кожній ознаці в залежності від її важливості та спроможності передавати інформацію. Математично, вагове об'єднання ознак можна виразити за допомогою формули:

$$F_{fused} = \sum_{i=1}^{n} w_i \cdot F_i$$

деF_{fused} - об'єднані ознаки,w_i- ваговий коефіцієнт для ознаки, F_i- окремі ознаки, n - кількість ознак.

Пізнє об'єднання ознак (late feature fusion) відбувається на пізніших етапах обробки, коли окремі ознаки з різних джерел або модальностей спочатку аналізуються окремо, а потім їх об'єднані репрезентації використовуються для рішення конкретних завдань. Цей підхід дозволяє зберігати більше контекстуальної інформації та використовувати більш гнучкі стратегії обробки.

У цьому підході, пізнє об'єднання ознак можна виразити за допомогою операторів агрегації, таких як середнє значення, максимальне значення або конкатенація. Наприклад, середнє значення ознак можна обчислити за допомогою формули:

$$F_{fused} = \frac{1}{n} \sum_{i=1}^{n} F_i$$

де F_fused- об'єднані ознаки, F_i - окремі ознаки, n - кількість ознак.

Завдяки використанню процесу об'єднання ознак та даних, що поєднують кілька джерел, можна досягти значного покращення продуктивності моделі. Цей процес сприяє покращенню якості розуміння даних та ефективному вилученню ознак, що своєю чергою сприяє покращенню загальної продуктивності та ефективності системи.

2.5. Відео ознаки

Відео ознаки (video features)- це числові характеристики відео, що використовуються для опису його властивостей. Вони включають різні типи ознак, які витягуються з відеоданих, щоб виокремити важливу інформацію та розпізнавати особливості відеосцен. Відео ознаки можуть бути засновані на різних аспектах відео, таких як кольорова інформація, рухові характеристики, форма об'єктів, текстура та багато інших.

Одним з найпоширеніших типів відео ознак є гістограми кольорів, які відображають розподіл кольорів у кадрі відео. Іншим типом ознак є рухові вектори, які відображають швидкість і напрямок руху об'єктів в кадрі. Також можуть використовуватись ознаки, що відображають форму об'єктів, текстуру, освітлення та інші аспекти відео.

Для отримання відео ознак з вхідного відео використовується спеціальний інструмент - відео екстрактор, який дозволяє видобувати різноманітні ознаки з відео. Відео екстрактори поділяються на два основні типи: модальні та мультимодальні. Модальний відео екстрактор отримує ознаки лише з відео, тоді як мультимодальний відео екстрактор може отримувати як відео ознаки, так і аудіо ознаки.

Використання відео екстрактора дозволяє отримати різноманітні ознаки з відео, такі як кольорові характеристики, рухові вектори, геометричні властивості, текстурні дескриптори та інші. Ці ознаки можуть бути використані для класифікації, розпізнавання об'єктів, відстеження руху та ін-

ших завдань аналізу відео. Додатково, мультимодальний відео екстрактор дозволяє поєднувати інформацію з відео та аудіо, що може покращити розуміння відеоданих та поліпшити результати аналізу.

В даному дослідженню було використано наявний відкритий набір даних зібраний в науковій роботі “Мультимодальне підсумовування створених користувачами відео” [2] (Multimodal summarization of user-generated videos) авторів: Псаллідас, Т., Коромілас, П., Джаннакопулос, Т., і Спіру, Е. у 2021 році.

Для обробки даних було використано мультимодальний відео екстрактор ознак з бібліотеки multimodal movie analysis. Цей інструмент приймає відео файл на вхід і видає отримані відео ознаки візуальних характеристик відео на виході. В контексті нашого дослідження, екстрактор видає 88 візуальних характеристик відео, за кожні 0,2 секунди.

Нижче наведено таблицю зі списком видобутих типів ознак та кількістю отриманих ознак з вхідного відео.

Характеристики кольору включають різноманітні показники, які відображають відтінки, насиченість та яскравість кольорів у відео. У нашому випадку ми використовуємо такі характеристики кольору:

— 8-бітна гістограма зелених значень: це розподіл кількості пікселів у відео за значеннями зеленого каналу, поділеними на 8 інтервалів.
— 8-бітна гістограма синіх значень: це розподіл кількості пікселів у відео за значеннями синього каналу, поділеними на 8 інтервалів.
— 8-бітна гістограма значень градацій сірого: це розподіл кількості пікселів у відео за значеннями градацій сірого кольору, поділеними на 8 інтервалів.
— 5-бітна гістограма максимального середньоквадратичного співвідношення для кожного триплету RGB: це розподіл кількості пікселів у відео за значеннями співвідношення максимального значення до середньоквадратичного значення для кожного триплету червоного,

Таблиця 2.1

Ознаки відео сигналу

Опис ознак	**Номер ознак**
Характеристики кольору: - 8-бітна гістограма червоних. - 8-бітна гістограма зелених значень. - 8-бітну гістограму синіх значень. - 8-бітну гістограму значень градацій сірого. - 5-бітна гістограма максимального середньоквадратичного співвідношення для кожного триплету RGB. - 8-бітну гістограму значень насиченості.	1-45
Середня абсолютна різниця між двома послідовними кадрами в градаціях сірого	46
Використання Viola-James методу в OpenCV бібліотеки для отримання: - Кількість виявлених облич - Середнє співвідношення площ обмежувальних прямокутників облич по відношенню до загальної площі кадру	47, 48
Оптичні характеристики руху за допомогою методу Лукаса-Кана: - Середня величина векторів потоку - Стандартне відхилення кутів векторів потоку - Співвідношення величини векторів потоку до відхилення кутів векторів потоку	49-51
Довжина зйомки кадру	52
Використання Single Shot Multibox Detector методу для отримання об'єктних ознак: - Метод включає підхід для виявлення 12 категорій об'єктів. - Для кожного виявленого об'єкта генеруються 3 статистичні показники (кількість виявлених об'єктів, середня впевненість виявлення та середнє співвідношення площі об'єктів до площі кадру).	52-88

зеленого і синього каналів, поділеними на 5 інтервалів.

— 8-бітна гістограма значень насиченості: це розподіл кількості пікселів у відео за значеннями насиченості кольору, поділеними на 8 інтервалів.

Характеристики кольору включають різноманітні показники, які відображають відтінки, насиченість та яскравість кольорів у відео. У нашому випадку ми використовуємо такі характеристики кольору: Середня абсолютна різниця між двома послідовними кадрами в градаціях сірого вимірює ступінь зміни піксельних значень між двома послідовними кадрами. Вона розраховується як середнє арифметичних значень абсолютних різниць піксельних значень у градаціях сірого між кожною відповідною парою пікселів. Метод Віоли-Джонса використовується для визначення кількості виявлених облич. Він базується на використанні класифікаторів, які аналізують різні частини зображення для виявлення облич. Результатом є кількість облич, виявлених на зображенні.

Метод Лукаса-Канаде використовується для отримання векторів потоку оптичних ознак. Він вимірює рух пікселів між двома послідовними кадрами та розраховує вектори, які відображають напрямок і величину цього руху.

Тривалість зйомки відноситься до тривалості відео, вимірюваної у кадрах або секундах. Це показник, який вказує на тривалість відео або конкретного сегмента відео.

Метод Single Shot Multibox Detector використовується для отримання об'єктних ознак, таких як категорії об'єктів. Він використовує нейронну мережу для виявлення та класифікації об'єктів на зображенні. У нашому випадку, цей метод використовується для виявлення 12 категорій об'єктів, таких як людина, транспортний засіб, вуличний краєвид, тварина, аксесуар, спортивне обладнання, кухонне приладдя, їжа, меблі, електроніка, побутова техніка та приміщення. Для кожного виявленого об'єкта також

генеруються 3 статистичні метрики: кількість виявлених об'єктів, середня впевненість виявлення та середнє співвідношення площі об'єктів до площі кадру.

2.6. Аудіо ознаки

Аудіо ознаки (audio features) - це числові представлення звукового сигналу, які використовуються для опису і аналізу аудіо даних. Вони дозволяють виявити і виміряти різні характеристики звуку, такі як акустичні властивості, часові та частотні характеристики, тембр, ритм і багато інших аспектів звукового сигналу.

В наборі даних, який використовується в дослідженні, для кожного аудіо кліпу застосовується pyAudioAnalysis бібліотека. Завдяки використанню цієї бібліотеки, екстракція аудіо ознак спочатку проводиться на основі короткострокового аналізу. На наступному етапі обчислюються статистичні характеристики ознак на рівні сегментів, які утворюють кінцеве представлення сегмента. Зокрема, аудіо сигнал розбивається на сегменти з використанням вікон короткострокового аналізу (з перекриттям або без перекриття), і для кожного сегменту проводиться короткострокова обробка, під час якої обчислюється 68 короткострокових ознак (34 ознаки та 34 дельти) для кожного короткострокового вікна. Короткострокові вікна зазвичай змінюються від 10 до 200 мілісекунд, тоді як сегментні вікна можуть бути від 0,5 секунди до декількох секунд, залежно від того, що вважається однорідним сегментом в конкретній області застосування. Короткострокові ознаки, які вилучаються використовуваною бібліотекою, поділяються на три категорії: часовий домен, частотний домен та кепстральний домен. Нижче наведена таблиця з назвами обраних ознак. Table 2.2

Згідно з описаною процедурою, для кожного аудіо-сегменту виконується витягування послідовності 68-мерних векторів ознак для кожного ко-

Таблиця 2.2

Ознаки звукового сигналу

Номер	**Назва**	**Опис ознаки**
1	Кількість перетинів нуля (Zero Crossing Rate)	Кількість перетинів нуля у звуковому сигналі, що вказує на зміну напрямку сигналу.
2	Енергія (Energy)	Сума квадратів значень сигналу, нормалізована до довжини кадру.
2	Ентропія енергії (Entropy of Energy)	Міра невизначеності енергії звукового сигналу, що відображає рівень варіаційності його енергетичних значень.
4	Спектральний центроїд (Spectral Centroid)	Центральна частота спектра звукового сигналу, яка відображає його загальну частотну характеристику.
5	Спектральне розподіл (Spectral Spread)	Міра розподілу частот в спектрі звукового сигналу, що вказує на ширину спектрального контенту.
6	Спектральна ентропія (Spectral Entropy)	Міра невизначеності спектру звукового сигналу, що відображає його рівень різноманітності частотних складових.
7	Спектральний потік (Spectral Flux)	Квадрат різниці між нормалізованими амплітудами спектрів двох послідовних кадрів.
8	Спектральний спад (Spectral Rolloff)	Частота, на яку припадає 90% розподілу амплітуд спектру
9-21	Кепстральні коефіцієнти частоти Мела (MFCCs)	Репрезентує собою кепстральні відображення звукового сигналу у частотному домені з використанням частотних смуг, масштабованих за шкалою Мела.
22-33	Вектор хроматизації (Chroma Vector)	Представляє собою 12-елементне представлення спектральної енергії в 12 рівномірно темперованих тональностях класів музики західного типу
34	Хроматичне відхилення (Chroma Deviation)	Стандартне відхилення 12 коефіцієнтів кольоровості.

роткочасового вікна. Ці вектори використовуються для обчислення статистичних характеристик на рівні сегменту, що представляють остаточне представлення сегменту. Конкретно, для кожного сегменту, який містить кілька короткочасових вікон з відповідними 68-мерними векторами ознак, обчислюються дві статистичні характеристики на рівні сегменту - середнє значення та стандартне відхилення. В результаті, загалом для кожного аудіо-сегменту використовуються 136 аудіо-статистик для його представлення.

2.7. Оцінка важливості ознак

У сфері машинного навчання ми стикаємося з викликом, відомим як проблема "Чорного ящика". Ця проблема виникає у контексті моделей, які не піддаються простій інтерпретації, аналізуючи лише їх параметри. Характерним прикладом є глибокі нейронні мережі, що мають значну кількість шарів та параметрів, і тому є типовими "чорними ящиками". Попри те, що вони можуть надавати високу точність прогнозування, їх внутрішню структуру та процес прийняття рішень складно інтерпретувати.

Проте, існують модельно-агностичні методи, які дозволяють зрозуміти, на основі яких причин модель прийшла до певних висновків, незалежно від конкретної моделі, що була використана. Ці методи можуть бути застосовані до майже будь-якої моделі машинного навчання після її навчання.

Прикладами таких методів є LIME (Local Interpretable Model-Agnostic Explanations) та SHAP (SHapley Additive exPlanations). LIME генерує спрощені моделі навколо прогнозів, щоб пояснити, як модель працює в околиці конкретного прикладу. SHAP, що базується на теорії ігор, використовує значення Шеплі для оцінки впливу кожного атрибуту на прогноз моделі. Це дозволяє нам оцінити важливість окремих ознак, що входять до моделі, та їх вплив на кінцевий результат.

2.8. Машинне навчання

Машинне навчання, як відгалуження штучного інтелекту, зосереджується на розробці алгоритмів та моделей, які здатні вчитися на основі даних, виконувати прогнози, класифікацію або приймати рішення без явного програмування. Термін "Машинне навчання"використовується, оскільки моделі та алгоритми набувають знань з досвіду або даних, які передаються комп'ютеру, а не в результаті прямого програмування.

Машинне навчання здобуло широку популярність завдяки своїм потенційним можливостям у різноманітних сферах. Центральна концепція полягає в тому, що моделі навчаються на великій кількості даних і виявляють закономірності, що дозволяють їм робити прогнози чи приймати рішення на основі нових, раніше невідомих даних. Це сприяє автоматизації процесів прийняття рішень, розпізнавання образів, категоризації даних та інших процесів.

Однією з ключових переваг машинного навчання є здатність моделей виявляти складні залежності та шаблони, які можуть бути важко або неможливо визначити аналітично. Вони можуть адаптуватися до змін у вхідних даних та вдосконалюватися з часом, що дозволяє їм забезпечувати кращі результати з кожним новим навчанням.

Машинне навчання можна розглядати через декілька основних типів:

1. Навчання з вчителем (Supervised Learning): В цьому контексті, моделі навчаються на основі позначених даних, де кожному вхідному зразку відповідає відома вихідна мітка або клас. Головною метою є побудова моделі, яка здатна робити точні прогнози для нових, раніше невідомих даних.
2. Навчання без вчителя (Unsupervised Learning): В цьому випадку, моделі навчаються на непозначених даних, де відсутні вихідні мітки або класи. Основною метою є виявлення прихованих законо-

мірностей, групування даних за схожістю або визначення основних характеристик даних.

3. Навчання з підкріпленням (Reinforcement Learning): В цьому контексті, моделі вдосконалюються через безпосередню взаємодію з динамічним середовищем, де вони отримують позитивні або негативні відгуки базуючись на діях. Основна мета полягає в максимізації сумарної винагороди та виборі оптимальної стратегії поведінки.

2.9. Навчання з вчителем (Supervised Learning)

"Навчання з вчителем"(Supervised Learning) є одним із основних підходів у галузі машинного навчання. Його головна ідея полягає в тому, що модель навчається на підготовлених тренувальних даних, де кожен зразок має відповідну мітку або класифікацію. Тренувальні дані складаються з вхідних зразків та їх відповідних міток, які відображають коректні відповіді або класифікацію для цих зразків.

Процес навчання з вчителем полягає у тому, щоб модель знаходила залежності та закономірності у тренувальних даних, що допомагають прогнозувати або класифікувати нові, раніше невідомі дані. Це досягається за допомогою різних алгоритмів та методів, таких як лінійна регресія, дерева рішень, нейронні мережі, метод опорних векторів та інші.

Процес навчання полягає у визначенні оптимальних параметрів моделі, які максимально відповідають тренувальним даним та їх міткам. Це включає мінімізацію помилок між прогнозами моделі та правильними відповідями. Після завершення процесу навчання, модель може бути використана для прогнозування або класифікації нових, раніше невідомих даних.

Навчання з вчителем має широкі застосування у різних галузях, включаючи обробку природних мов, комп'ютерний зір, фінанси, медицину та інші. Цей підхід дозволяє вирішувати завдання прогнозування, класифікації,

розпізнавання образів, аналізу даних та інших, що робить його надзвичайно популярним та потужним інструментом у галузі машинного навчання та штучного інтелекту.

2.10. Випадковий ліс (Decision Forest)

2.10.1. Дерево рішень (Decision Tree).

Дерево рішень являє собою потужний та гнучкий алгоритм машинного навчання, який може бути застосований для вирішення завдань класифікації та регресії. Цей алгоритм є ключовим елементом моделей випадкового лісу, які відносяться до найбільш стабільних та широко використовуваних моделей в сучасному машинному навчанні.

Однією з основних переваг дерев рішень є їх прозорість та інтерпретованість. На відміну від багатьох інших моделей, дерева рішень можна візуалізувати та інтуїтивно розуміти, що робить їх відмінним інструментом для експлоративного аналізу даних. Вони здатні обробляти як числові, так і категоріальні дані, і не вимагають великої кількості попередньої обробки даних, такої як нормалізація або стандартизація.

Центральна концепція дерев рішень полягає в принципі рекурсивного розбиття. Починаючи з кореневого вузла, що включає весь набір даних, дані розбиваються на менші підмножини, які стають все більш однорідними відносно цільової змінної. Це розбиття здійснюється на основі простих правил рішень, які формуються на основі характеристик даних.

Основні елементи дерева рішень:

1. Вузли (Nodes): Ці елементи являють собою правила прийняття рішень. Кожен вузол аналізує конкретний атрибут та розбиває дані відповідно до його значення. Верхній вузол, що містять весь набір даних, відомий як кореневий вузол.

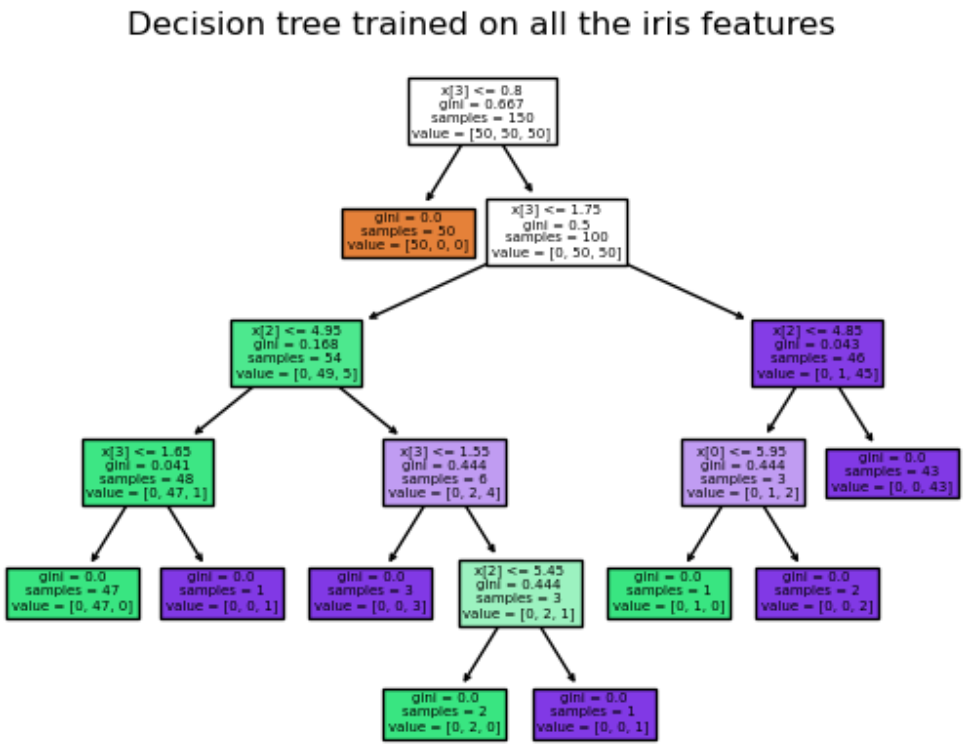
2. Ребра (Edges): Ці елементи відображають результати аналізу вузла і

з'єднують вузли всередині дерева.

3. Листя (Leaves): Це термінальні вузли, які передбачають результат (мітку класу або неперервне значення).

Приклад використання дерев наведено на наступному рисунку 2.7:

Рис. 2.1. Приклад використання дерева [23]



Основні поняття, що стосуються дерев рішень:

— **Розбиття (Splitting)**: Це процес поділу даних на підмножини на основі критерію, застосованого до певного атрибута.

— **Обрізка (Pruning)**: Це процес видалення надлишкових вузлів з дерева рішень з метою підвищення його ефективності та запобігання перенавчанню.

— **Гілка (Branch)**: Це секція дерева, яка починається в певному вузлі

і включає всі можливі наслідки тесту, проведеного в цьому вузлі.

— **Глибина (Depth)**: Це відстань від кореневого вузла до найвіддаленішого листа.

Дерева рішень можуть бути застосовані для вирішення широкого спектра задач. Вони особливо ефективні в ситуаціях, коли важлива інтерпретованість. Вони можуть обробляти як бінарні, так і багатокласові задачі класифікації, а також задачі регресії.Також використовуються в ансамблевих методах для підвищення точності прогнозування.

Тепер давайте більш детально розглянемо ключові концепції, що стосуються дерев рішень:

— **Ентропія (Entropy)**: Це статистична міра, яка використовується для визначення ступеня невизначеності або випадковості в наборі даних. У контексті дерев рішень, ентропія використовується для визначення "чистоти"вузлів. Чим вища ентропія, тим більше невизначеності або змішування в даних, і навпаки.

— **Приріст інформації (Information Gain)**: Це метод визначення, який атрибут найкраще розділяє набір даних. Приріст інформації вимірює зменшення ентропії після розбиття набору даних на підмножини. Атрибут з найбільшим приростом інформації вибирається для розбиття.

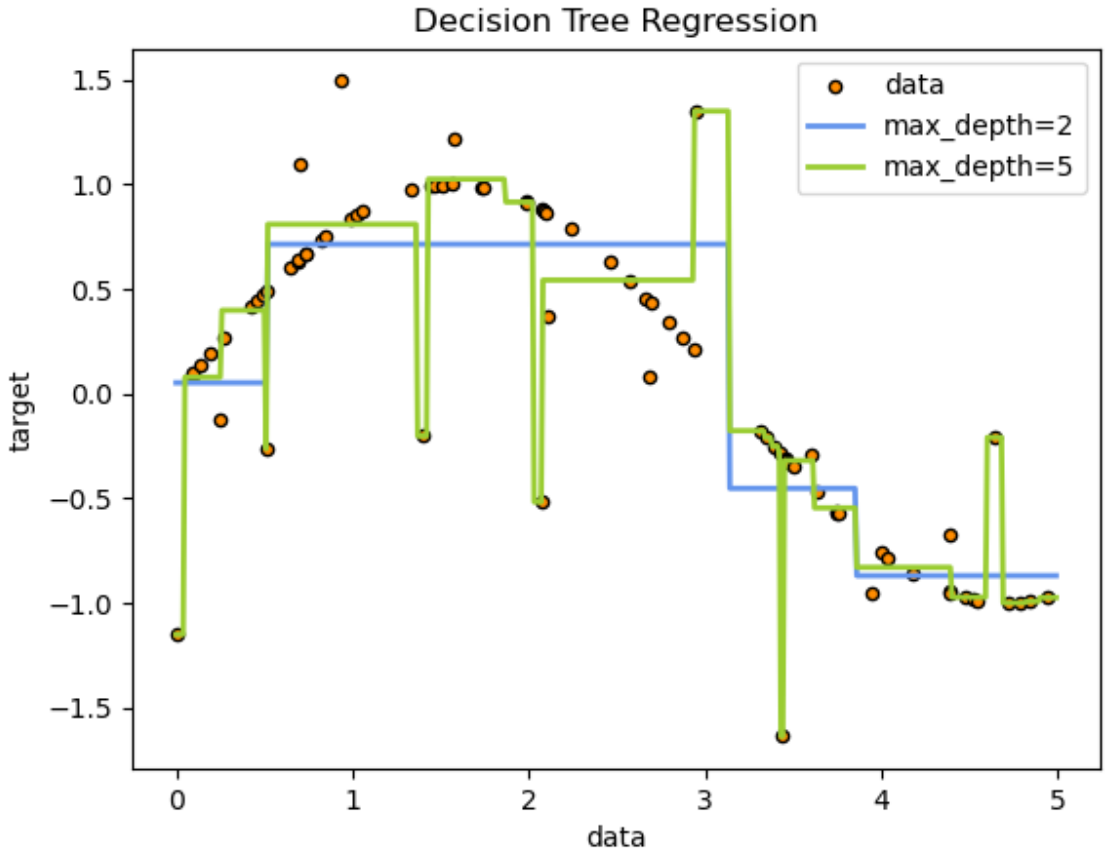
— **Індекс Джіні (Gini Impurity)**: Це альтернативний до ентропії метод вимірювання "чистоти"вузла. Індекс Джіні вимірює ймовірність помилкової класифікації елемента, вибраного випадковим чином, якщо йому присвоїти мітку на основі розподілу міток в підмножині.

— **Перенавчання (Overfitting)**: Це ситуація, коли модель занадто "підлаштовується"під навчальні дані, внаслідок чого втрачає здатність до узагальнення на нових даних. В контексті дерев рішень, перенавчання може виникнути, коли дерево занадто глибоке, що

призводить до створення складних правил, які відображають шум або випадковості в навчальних даних. Методи обрізки дерева можуть бути використані для запобігання перенавчанню, обмежуючи глибину дерева або встановлюючи мінімальну кількість прикладів у вузлі перед його розбиттям.

Приклад використання дерева рішення для задачі регресії з параметром глибини, який дорівнює 2 або 5, наведено на наступному рисунку 2.2:

Рис. 2.2. Приклад використання дерева 2 [23]



2.10.2. Часова складність алгоритму дерева рішень.

Часова складність побудови дерева рішень є функцією від кількості записів та кількості особливостей у заданих даних.

У найгіршому випадку часова складність побудови дерева рішень становить

$$O(N \cdot M \cdot \log(N))$$

, де: - N - кількість записів (або екземплярів) у навчальних даних. - M - кількість особливостей (або атрибутів) в даних.

Ця складність виникає тому, що для кожного вузла дерева алгоритм ітерується через кожен запис в даних (що займає $O(N)$ часу) і перевіряє кожну особливість для обчислення найкращого розбиття (що займає $O(M)$ часу). Цей процес повторюється для кожного з N записів, і оскільки записи сортуються перед обчисленням найкращого розбиття, додатковий фактор $O(log(N))$ вводиться для кроку сортування.

Важливо зауважити, що це часова складність для побудови дерева рішень. Часова складність для роботи прогнозів з деревом рішень становить $O(log(N))$, оскільки процес прогнозування включає проходження дерева від кореня до листа, і добре збалансоване бінарне дерево має логарифмічну висоту.

Однак ці складності можуть варіюватися в залежності від специфіки використовуваного алгоритму (наприклад, ID3, C4.5, CART тощо), якості даних та конфігурації дерева (наприклад, максимальна глибина та мінімальні зразки розбиття).

При зміні сценаріїв, можна побачити, як буде змінюватися формула:

1. **Якщо кількість особливостей фіксована**: Якщо кількість особливостей M є фіксованою, тоді часова складність побудови дерева рішень стає $O(N \cdot \log(N))$. Це означає, що час побудови зростає логарифмічно з кількістю записів.
2. **Якщо кількість записів фіксована**: Якщо кількість записів N є фіксованою, тоді часова складність побудови дерева рішень стає $O(M)$. Це означає, що час побудови зростає лінійно з кількістю осо-

бливостей.

3. **Якщо дерево рішень є повним бінарним деревом**: Якщо дерево рішень є повним бінарним деревом, тоді висота дерева h є $O(\log(N))$, і часова складність для роботи прогнозів стає $O(\log(N))$.
4. **Якщо дерево рішень є зігнутим (skewed)**: Якщо дерево рішень є зігнутим, тобто одна гілка дерева значно довша за інші, тоді висота дерева h може бути $O(N)$, і часова складність для роботи прогнозів також стає $O(N)$.

Ці приклади показують, що часова складність побудови дерева рішень та роботи прогнозів може суттєво змінюватися в залежності від різних факторів.

2.10.3. Алгоритми для побудови дерев рішень.

Iterative Dichotomiser 3 (ID3): ID3, створений Россом Квінланом у 1986 році, був одним з перших алгоритмів, що використовували концепцію приросту інформації, заснованої на ентропії, для вибору атрибутів для розбиття. Проте, ID3 має декілька обмежень. Він може працювати лише з категоріальними атрибутами, не вміє обробляти пропущені значення, і не має вбудованої стратегії обрізки, що може призвести до перенавчання. Ці обмеження зробили ID3 менш практичним для багатьох реальних застосувань.

C4.5: Квінлан продовжив роботу над ID3, що призвело до створення алгоритму C4.5. Він вніс кілька значних покращень до ID3, включаючи здатність обробляти числові атрибути за допомогою концепції розбиття на інтервали. C4.5 також може обробляти пропущені значення, використовуючи різні стратегії, і включає механізм обрізки для боротьби з перенавчанням. Ці покращення зробили C4.5 більш практичним і широко використовуваним алгоритмом.

C5.0: C5.0, що є останньою версією серії алгоритмів Квінлана, вніс ряд покращень до C4.5. Він є швидшим і ефективнішим за пам'яттю, ніж C4.5, і включає покращення для обробки пропущених значень і підтримки бустінгу, техніки, яка може значно покращити точність прогнозування.

Classification and Regression Trees (CART): CART, розроблений Брейманом, Фрідманом, Олшеном та Стоуном у 1984 році, є ще одним важливим алгоритмом для побудови дерев рішень. Він може використовуватися як для задач класифікації, так і для задач регресії, що робить його універсальним інструментом. CART використовує метрику, відому як "Індекс Джіні для вибору атрибутів для розбиття. Особливістю CART є те, що він побудовує бінарні дерева, тобто кожен вузол має рівно два дочірні вузли, що відрізняє його від ID3, C4.5 та C5.0, які можуть побудувати дерева з більш ніж двома дочірніми вузлами.

Всі ці алгоритми можуть працювати з категоріальними даними, але тільки C4.5, C5.0 та CART можуть безпосередньо обробляти числові дані. ID3 може бути адаптований для роботи з числовими даними шляхом перетворення числових значень на категоріальні, але це може бути неефективно для даних з великою кількістю унікальних числових значень.

Історія розвитку цих алгоритмів показує постійне покращення та адаптацію до змінних вимог і обмежень. Від простого ID3, що має обмежену здатність до обробки різних типів даних та проблеми з перенавчанням, до більш складних і гнучких алгоритмів, таких як C4.5, C5.0 та CART, що можуть обробляти різні типи даних, пропущені значення, і мають стратегії для боротьби з перенавчанням.

2.10.4. Об'єднання ознак.

Якщо взяти набір навчальних векторів $x_i \in R^n, \mathrm{i} = 1, \ldots, |$, де R^n це n-вимірний векторний простір, та вектор міток $y \in R^l$, де R^l це l-вимірний векторний простір, дерево рішень виконує систематичний розподіл про-

стору ознак. Це означає, що воно використовує рекурсивний процес, щоб розділити простір на підпростори так, щоб зразки з однаковими мітками або схожими цільовими значеннями опинилися в одному підпросторі.

Нехай, дані в вузлі m будуть визначенні як Q_m з n_m зразками. Тоді Q_m - це набір зразків в вузлі, а n_m - це кількість зразків в цьому вузлі. Для кожного потенційного розбиття $\theta = (j, t_m)$, де j - буде ознакою, а t_m буде порогом, де дані розбиваються на підмножини $Q_m^{\text{left}}(\theta)$ та$Q_m^{\text{right}}(\theta$.

$$Q_m^{\text{left}}(\theta) = \{(x, y) \mid x_j \leqslant t_m\}$$
$$Q_m^{\text{right}}(\theta) = Q_m \backslash Q_m^{\text{left}}(\theta)$$

Вище зазначені формули визначають, як дані будуть розбиватися на підмножини. $Q_m^{\text{left}}(\theta)$ - це підмножина зразків, де значення ознаки j менше або дорівнює порогу t_m. $Q_m^{\text{right}}(\theta)$ - це підмножина зразків, що залишилися після вибору $Q_m^{\text{left}}(\theta)$.

Якість потенційного розбиття вузла m визначається за допомогою функції невизначеності або функції втрат $H()$, вибір якої залежить від типу задачі (класифікація або регресія). Функція невизначеності, також відома як функція ентропії, вимірює ступінь невизначеності в даних. Вона використовується для визначення "чистоти"вузла в дереві рішень. Функція втрат, з іншого боку, вимірює відхилення прогнозованих значень від дійсних значень.

$$G(Q_m, \theta) = \frac{n_m^{left}}{n_m} H\left(Q_m^{left}(\theta)\right) + \frac{n_m^{\text{right}}}{n_m} H\left(Q_m^{\text{right}}(\theta)\right)$$

Ця формула визначає якість потенційного розбиття. $G(Q_m, \theta)$ це загальна невизначеність після розбиття, яка є зваженою сумою невизначеностей лівої та правої підмножин. n_m^{left} та n_m^{right} - це кількість зразків в лівій та правій підмножинах відповідно.

Для того, щоб обрати параметри, які мінімізують невизначеність, використовується наступна формула:

$$\theta^* = \text{argmin}_\theta \, G\left(Q_m, \theta\right)$$

Ця формула визначає оптимальні параметри розбиття. Де θ^* - це параметри розбиття, які мінімізують загальну невизначеність.

Наступним кроком йде рекурсивне проходження підмножин $Q_m^{\text{left}}\ (\theta^*)$ та $Q_m^{\text{right}}\ (\theta^*)$ до того моменту, коли буде досягнуто максимально встановлена глибина, $n_m < \min_{\text{samples}}$ або $n_m = 1$. Це означає, що процес продовжується для кожного вузла, поки не буде досягнута одна з вище наведених умов.

Критерії класифікації є основою для будь-якого алгоритму класифікації, включаючи дерева рішень. Вони визначають, як ми розділяємо наші дані на групи або класи. Це важливо, тому що ми хочемо, щоб наші класифікаційні моделі були якомога точнішими і робили якомога менше помилок. Для цього ми використовуємо різні критерії, щоб визначити, як найкраще розділити наші дані. Два з найпоширеніших критеріїв - це індекс Джині та ентропія, які ми обговорюємо далі.

$$p_{mk} = \frac{1}{n_m} \sum_{y \in Q_m} I(y = k)$$

- p_{mk} - це пропорція спостережень класу k у вузлі m. Це допомагає нам зрозуміти, яка частина спостережень у даному вузлі належить до певного класу. - n_m - це кількість спостережень у вузлі m. Це загальна кількість спостережень, які ми розглядаємо в даному вузлі. - $\sum_{y \in Q_m} I(y = k)$- це сума індикаторних функцій, які дорівнюють 1, якщо $y = k$, і 0 в іншому випадку, для всіх y у Q_m, де Q_m - це набір спостережень у вузлі m.

Ця формула використовується для обчислення пропорції спостережень кожного класу в кожному вузлі.

$$H\left(Q_m\right) = \sum_k p_{mk} \left(1 - p_{mk}\right)$$

- $H(Q_m)$- це індекс Джині для вузла m.

$$\sum_k p_{mk} \left(1 - p_{mk}\right)$$

- це сума добутків пропорцій спостережень класу k та $1 - p_{mk}$ для всіх класів k.

Індекс Джині вимірює ймовірність того, що дві випадково вибрані спостереження з вузла належатимуть до різних класів. Індекс Джині варіюється від 0 до 1, де 0 відповідає "чистому"вузлу (всі спостереження належать до одного класу), а 1 відповідає найбільшій невизначеності (спостереження рівномірно розподілені між класами).

$$H\left(Q_m\right) = -\sum_k p_{mk} \log\left(p_{mk}\right)$$

- $\sum_k p_{mk} \log\left(p_{mk}\right)$ - це сума добутків пропорцій спостережень класу k та логарифму пропорції спостережень класу k для всіх класів k, зі знаком мінус.

Ця формула використовується для обчислення ентропії. Ентропія, як і індекс Джині, використовується для визначення якості розбиття в дереві рішень. Ентропія вимірює "непорядок"або "хаос"в даних. Чим вище ентропія, тим більше невизначеності або "нечистоти"в даних. Ентропія вимірюється в одиницях, які називаються "джоулі на кельвін"(J/K), і вона використовується в термодинаміці для вимірювання ступеня непорядку або хаосу в системі. У контексті дерев рішень, висока ентропія вказує на високий рівень невизначеності або змішування класів у вузлі.

2.10.5. Ансамблеві методи (Ensembles).

Ансамблеві методи являють собою стратегію машинного навчання, яка об'єднує декілька моделей з метою підвищення точності прогнозування. Ця концепція базується на принципі, що комбінація, або "ансамбль моде-

лей може створити сильну модель, навіть якщо окремі моделі є слабкими. Це стає можливим завдяки тому, що різні моделі можуть виявляти різні закономірності в даних, а їх комбінування може дати повніше розуміння даних.

У контексті дерев рішень, ансамблеві методи використовуються для побудови кількох дерев рішень та їх подальшого об'єднання. Це допомагає зменшити варіативність (overfitting), яка часто виникає при використанні одного дерева рішень, та підвищує стабільність прогнозування.

Існують три основні типи ансамблевих методів, які використовуються в контексті дерев рішень: бустінг, беггінг та стекінг.

1. Бустінг (*Boosting*) - це метод, який використовує послідовність моделей, де кожна наступна модель намагається виправити помилки попередньої. Одним з найвідоміших алгоритмів бустінгу є алгоритм градієнтного бустінгу.
2. Беггінг (*Bootstrap Aggregating*) - це метод, який використовує паралельну послідовність моделей, кожна з яких навчається на випадковій підмножині даних. Результати цих моделей потім усереднюються. Найвідомішим алгоритмом беггінгу є випадковий ліс (*Random Forest*).
3. Стекінг (*Stacking*) - це метод, який використовує моделі різних типів, а потім використовує ще одну модель (зазвичай називається мета-моделлю), щоб комбінувати їх прогнози.

Ансамблеві методи використовуються з кількох причин. По-перше, вони дозволяють покращити точність прогнозування. По-друге, вони допомагають уникнути перенавчання, оскільки комбінація моделей часто менш схильна до перенавчання, ніж окрема модель. По-третє, вони дозволяють використовувати різні типи моделей, що може бути корисним, коли немає однозначної відповіді на питання, який тип моделі найкраще використовувати.

Все це робить ансамблеві методи дуже потужним інструментом в машинному навчанні. Вони вже довели свою ефективність в різних задачах, включаючи класифікацію, регресію і ранжування, і є одними з найпопулярніших методів в області машинного навчання.

2.10.6. Випадковий ліс (Random Decision Forest).

Випадковий ліс (Random Forest) є одним із найпопулярніших типів ансамблевих методів, який використовується для задач класифікації і регресії. Основна ідея Random Forest полягає в поєднанні багатьох рішаючих дерев в один ансамбль, де кінцеве рішення приймається шляхом більшості голосів (Majority Wins).

Процес побудови Random Forest починається зі створення випадкових підвибірок з навчального набору даних (bootstrap sampling). Кожна підвибірка створюється шляхом вибору випадкових прикладів заміщенням, що означає, що один і той самий приклад може з'явитися в підвибірці більше одного разу, тоді як інші приклади можуть бути пропущені. Це забезпечує різноманітність у моделях.

Потім для кожної підвибірки будується окреме дерево прийняття рішень. При побудові кожного дерева на кожному вузлі замість розгалужень усі змінні не розглядаються, але лише певна підмножина змінних випадковим чином вибирається. Цей процес називається випадковим підпростором (random subspace).

Після побудови всіх дерев Random Forest використовується для класифікації або регресії нових прикладів. Коли потрібно зробити прогноз, кожне дерево вносить свої прогнози, і за допомогою голосування більшості голосів (або середнього значення для регресії) визначається кінцевий прогноз ансамблю.

"Випадковий ліс"(Random Forest) відзначається рядом переваг, які сприяють його широкому застосуванню в різних сферах:

1. Стійкість до перенавчання: Випадковий ліс, завдяки використанню випадкових підвибірок та випадкових підпросторів, має властивість бути менш схильним до перенавчання, порівняно з окремим деревом. Це забезпечує більшу надійність моделі та зменшує ризик переоснащення.
2. Висока точність прогнозування: Випадковий ліс здатний досягати високої точності прогнозування, особливо в задачах класифікації. Це досягається за рахунок голосування багатьох дерев, що дозволяє отримати більш точний та надійний прогноз.
3. Здатність працювати з великими наборами даних: Випадковий ліс може ефективно обробляти великі набори даних. Він має низьку обчислювальну складність порівняно з іншими складними моделями, що робить його вибором номер один для великих наборів даних.
4. Вбудована оцінка важливості змінних: Випадковий ліс надає оцінку важливості змінних, що дозволяє визначити, які змінні мають найбільший вплив на прогнозування. Це допомагає зрозуміти, які фактори найбільше впливають на результати моделі, та спрямувати увагу на них при подальшому аналізі.

2.11. Навчання без вчителя (Unsupervised Learning)

Навчання без вчителя (Unsupervised Learning) є одним з ключових підходів в області машинного навчання. Відрізняючись від навчання з вчителем, цей підхід передбачає навчання моделі на основі непозначених даних, без використання правильних відповідей або міток. Основна мета полягає в тому, щоб модель самостійно виявляла структуру, закономірності або групи в даних без явного навчання на основі міток.

У рамках навчання без вчителя, модель самостійно виявляє приховані закономірності та структуру даних за допомогою розробки алгоритмів кла-

стеризації, виявлення асоціативних правил, виявлення аномалій та інших методів. Кластеризація є одним з основних завдань навчання без вчителя, де модель групує подібні об'єкти разом на основі їх схожості.

Навчання без вчителя має широкий спектр застосувань. Воно може використовуватися для виявлення нових шаблонів, сегментації даних, скорочення розмірності, рекомендаційних систем, аналізу соціальних мереж та багатьох інших областей. Цей підхід дозволяє моделям самостійно виявляти внутрішні залежності і структуру в даних, що дає можливість отримувати цінні висновки та розуміння без потреби в явному навчанні на основі міток.

2.12. Метод k-найближчих сусідів (KNN)

Метод k-найближчих сусідів (KNN) - це алгоритм навчання з учителем, який широко використовується для вирішення задач класифікації та регресії.

Робота KNN базується на простому принципі: для класифікації нового прикладу алгоритм шукає k найближчих сусідів до цього прикладу, використовуючи певну метрику відстані, і визначає його клас на основі більшості класів цих сусідів. Для задач регресії алгоритм використовує середнє або медіанне значення цих сусідів для передбачення значення нового прикладу.

Основні варіанти застосування KNN:

1. Класифікація: В цьому випадку KNN визначає клас нового прикладу, шляхом голосування k найближчих сусідів. Клас, який має найбільшу кількість представників серед сусідів, стає прогнозом для нового прикладу.
2. Регресія: В регресійному KNN, алгоритм визначає значення нового прикладу, шляхом обчислення середнього (або медіанного) значен-

ня цільової змінної в k найближчих сусідів.

Нижче наведені малюнки з прикладами використання KNN в задачах класифікації 2.3 та регресії 2.4.

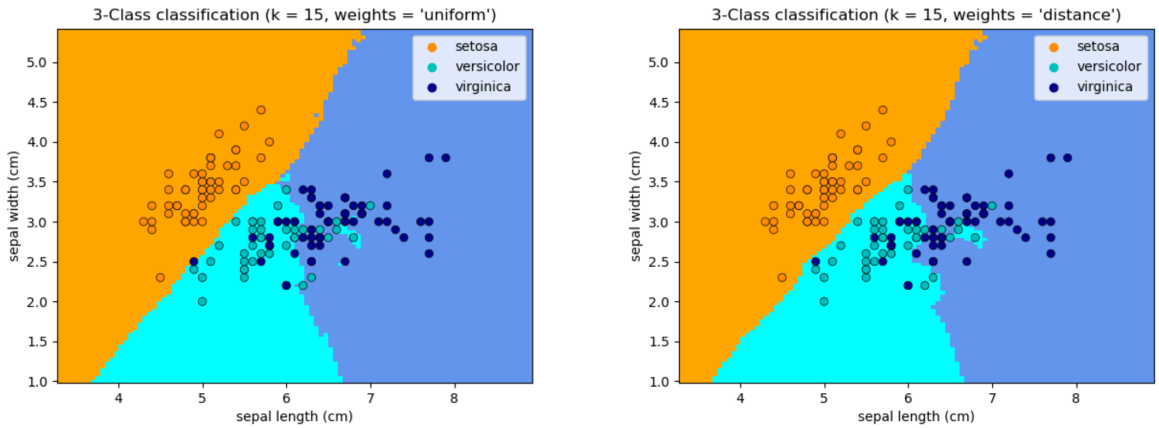


Рис. 2.3. Приклад використання KNN у задачі класифікації [22]

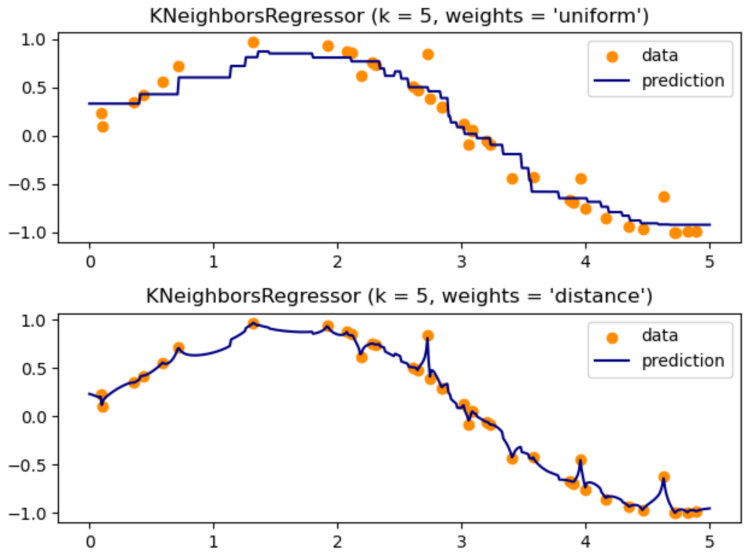


Рис. 2.4. Приклад використання KNN у задачі регресії [22]

Головний параметр KNN, який визначає кількість найближчих сусідів, - це k, що використовується для прийняття рішення. Значення k повинно бути вибране таким чином, щоб забезпечити оптимальний баланс між

зменшенням шуму в даних і уникненням високої варіабельності. Якщо k занадто мале, модель може стати чутливою до шуму, випадковостей та викидів в даних, що може призвести до перенавчання. З іншого боку, якщо k занадто велике, модель може стати менш чутливою до важливих відмінностей між класами, що може призвести до недостатнього навчання.

Функція відстані в KNN вказує на відстань між прикладами і використовується для визначення близькості між сусідами. Найпоширеніші функції відстані включають Евклідову, Манхеттенську, Чебишова та косинусну відстань. Важливо враховувати, що вибір функції відстані в KNN може суттєво вплинути на результати класифікації або регресії. Вибір відстані повинен враховувати особливості даних, такі як масштабування ознак, наявність викидів, типи ознак та контекст задачі.

Ось детальніше про кожен тип відстані:

1. Евклідова відстань: Це найбільш поширений тип відстані, визначений як відстань між двома точками у просторі. Формула Евклідової відстані:

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

.

2. Манхеттенська відстань: Ця відстань вимірює суму абсолютних різниць між координатами двох точок. Формула Манхеттенської відстані:

$$d = |x_2 - x_1| + |y_2 - y_1|$$

.

3. Чебишова відстань: Ця відстань визначається як максимальна абсолютна різниця між координатами двох точок. Формула Чебишової відстані:

$$d = \max(|x_2 - x_1|, |y_2 - y_1|)$$

.

4. Мінковська відстань: Це загальний тип відстані, який включає як Евклідову, так і Манхеттенську відстані як спеціальні випадки. Мінковська відстань визначається за допомогою параметра p, який визначає степінь абсолютної різниці між координатами двох точок. Формула Мінковської відстані:

$$d = \left(\sum_{i=1}^{n} |x_i - y_i|^p \right)^{1/p}$$

.

Коли $p = 2$, Мінковська відстань стає Евклідовою відстанню. Коли $p = 1$, Мінковська відстань стає Манхеттенською відстанню.

Мінковська відстань дозволяє нам вибирати між різними типами відстаней, залежно від конкретного контексту або вимог до задачі.

У виборі типу відстані важливо враховувати особливості даних та контекст задачі, оскільки вона може суттєво впливати на результат KNN алгоритму.

При реалізації алгоритму KNN існує кілька різних підходів до пошуку найближчих сусідів. Вибір конкретного алгоритму для пошуку залежить від розміру тренувального набору даних, кількості ознак та потреби в ефективності обчислень. Декілька найпоширеніших алгоритмів для пошуку включають "Brute Force "K-D Tree"та "Ball Tree".

Метод Brute Force, також відомий як наївний підхід, використовує метод перебору, щоб знайти k найближчих сусідів для нового зразка даних. Основна ідея полягає в тому, що для кожного нового зразка даних ми обчислюємо відстань між ним і всіма іншими зразками в навчальному наборі. Потім ми вибираємо k найближчих сусідів, які мають найменшу відстань до нового зразка. Ці сусіди визначають класифікацію або прогноз для нового зразка. Метод є простим, але може бути неефективним для великих наборів даних, оскільки вимагає обчислення багатьох відстаней. Крім того, він може бути чутким до шуму та нелінійності в даних.

K-вимірне дерево (k-dimensional tree) - це ефективний алгоритм для методу k-найближчих сусідів (KNN), який допомагає прискорити пошук найближчих сусідів. На кожному вузлі K-d дерева обирається змінна, за якою розбивається простір ознак. Наприклад 2.5, у двовимірному просторі ознак можна обрати змінну x або y для розбиття. Ми можемо розбити простір ознак таким чином, що створюємо дві гілки: одну, де значення обраної змінної менше порогового значення, і іншу, де значення вище порогу. Цей процес повторюється для кожного рівня вузлів, створюючи структуру дерева. Під час пошуку найближчих сусідів ми порівнюємо новий приклад з кожним вузлом дерева, обираючи більш обмежений набір точок для подальшого обчислення відстані. Це зменшує кількість обчислень порівнянь і полегшує пошук.

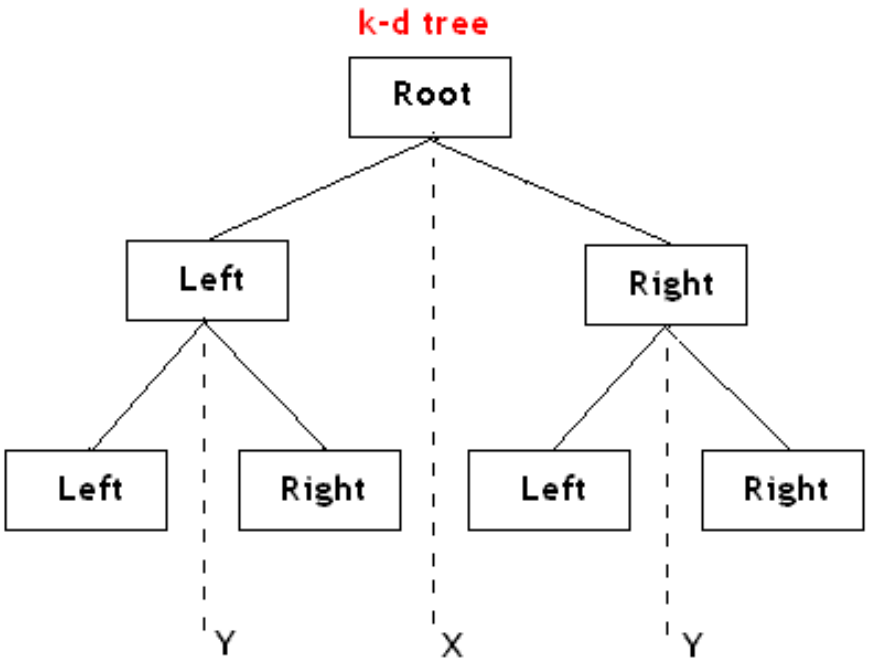


Рис. 2.5. Приклад K-D дерева [20]

Дерево куль 2.6, або Ball Tree, є альтернативним методом структуризації даних, який створює бінарне дерево, де кожен вузол представляє собою "кулю"або гіперсферу в просторі ознак, яка містить підмножину трену-

вальних прикладів. Кожна сфера містить центральну точку та радіус, який охоплює всі точки вузла.

Процес створення дерева куль починається з усієї тренувальної вибірки, яка обгортається мінімальною можливою кулею. Ця куля потім розбивається на дві менші кулі, кожна з яких обгортає свою власну підмножину тренувальних прикладів. Цей процес рекурсивно повторюється, доки кожна куля не містить лише один тренувальний приклад, формуючи листя вузлів дерева.

Під час пошуку найближчих сусідів для нового прикладу, алгоритм спочатку визначає, які кулі можуть містити найближчих сусідів, використовуючи геометричні властивості куль. Потім, він обчислює відстані лише до тренувальних прикладів, які знаходяться в цих кулях. Це значно зменшує кількість необхідних обчислень відстані, особливо для великих тренувальних наборів даних.

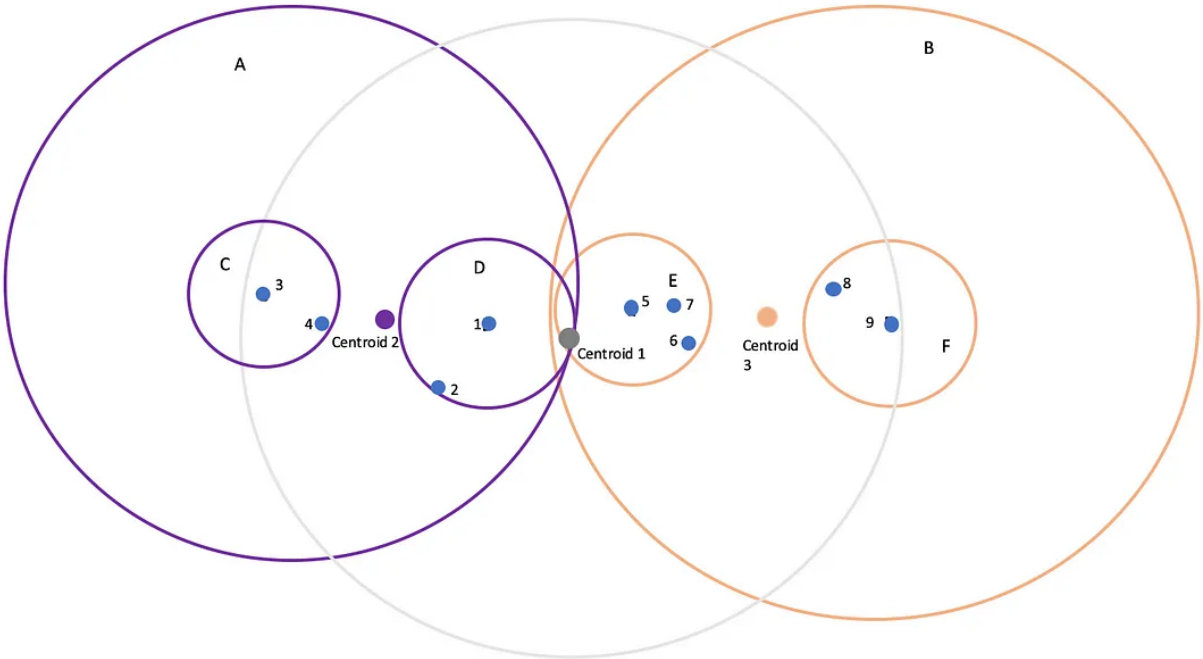


Рис. 2.6. Приклад дерева куль [19]

Метод k-найближчих сусідів (KNN) відзначається своєю простотою та

зрозумілістю при інтерпретації результатів. Рішення, що беруться на основі цього методу, можуть бути легко пояснені, оскільки вони ґрунтуються на конкретних сусідах у просторі ознак.

Наприклад, у випадку класифікації, можна вказати, які саме приклади вплинули на прийняте рішення. Це робить метод k-найближчих сусідів привабливим для застосування у випадках, коли важлива поясненість та зрозумілість результатів.

2.12.1. Математичне формулювання.

$$\arg\max_{L} \sum_{i=0}^{N-1} p_i$$

Елементи формули:

1. L - матриця лінійного перетворення розміру (n components, n features), яку ми намагаємося вивчити.
2. N - кількість зразків.
3. p_i - ймовірність правильної класифікації зразка i.
4. $\arg\max$ - оператор, який повертає значення аргумента, при якому функція досягає свого максимального значення.

Ця формула використовується для вивчення оптимальної матриці лінійного перетворення L, яка максимізує суму ймовірностей p_i правильної класифікації всіх зразків. Це є основною метою алгоритму NCA (Neighbourhood Components Analysis).

$$p_i = \sum_{j \in C_i} p_{ij}$$

Елементи формули:

- p_i - ймовірність правильної класифікації зразка i. - C_i - множина точок, що належать до того ж класу, що і зразок i. - p_{ij} - ймовірність, що зразок i правильно класифікується як зразок j.

Ця формула використовується для обчислення ймовірності p_i, яка є сумою ймовірностей p_{ij} для всіх зразків j, що належать до того ж класу, що і зразок i. Це допомагає визначити, наскільки добре зразок i класифікується за правилом найближчого сусіда в просторі, що вивчається.

$$p_{ij} = \frac{\exp\left(-\|Lx_i - Lx_j\|^2\right)}{\sum_{k \neq i} \exp - \left(\|Lx_i - Lx_k\|^2\right)}, \quad p_{ii} = 0$$

Елементи формули:

- p_{ij} - ймовірність, що зразок i правильно класифікується як зразок j.

- L - матриця лінійного перетворення, яку ми намагаємося вивчити. - x_i та x_j - зразки, які ми порівнюємо.

- $\|Lx_i - Lx_j\|^2$ - квадрат евклідової відстані між перетвореними зразками i та j.

- exp - експоненціальна функція, яка використовується для перетворення відстані в ймовірність.

- $\sum_{k \neq i} \exp - \left(\|Lx_i - Lx_k\|^2\right)$ - нормалізуючий член, який забезпечує, що сума ймовірностей для всіх зразків дорівнює 1.

- $p_{ii} = 0$ - зразок не може бути класифікований як самого себе, тому p_{ii} завжди дорівнює 0.

Ця формула використовується для обчислення ймовірності, що зразок j є найближчим сусідом зразка i в просторі, який був отриманий в результаті вбудовування. Використовується функція softmax, що перетворює відстані між вбудованими представленнями на ймовірності. Вона робить це шляхом застосування експоненціальної функції до від'ємного квадрату відстані і подальшого нормування цих значень так, що їх сума по всіх зразках, крім i, дорівнює 1.

Ця формула допомагає нам краще розуміти, як зразки розташовані відносно один одного в просторі, отриманому в результаті вбудовування, і як це впливає на їх класифікацію.

2.13. Метрики оцінювання (Evaluation Metrics)

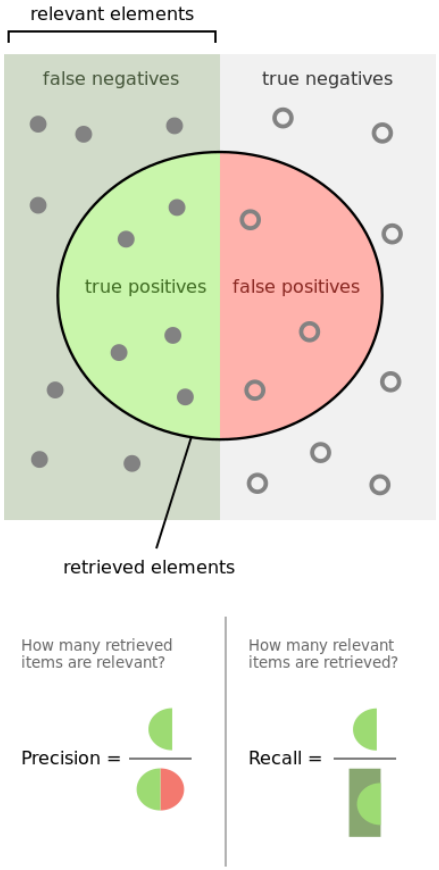
Метрики оцінювання в машинному навчанні використовуються для кількісного вимірювання ефективності моделей на основі їх прогнозів. Вони дозволяють об'єктивно порівняти різні моделі або налаштування моделей, вимірюючи різні аспекти моделі, такі як точність, повнота, специфічність, середня квадратична помилка, коефіцієнт детермінації тощо.

Основні категорії метрик включають:

1. Метрики класифікації: Використовуються для оцінки якості моделей у задачах класифікації, де потрібно призначити кожному прикладу певний клас. Деякі популярні метрики класифікації включають точність (*accuracy*), влучність (*precision*) 2.7, повноту (*recall*), F-міра (F-*score*), AUC-ROC (площа під кривою ROC) та матрицю помилок (*confusion matrix*).
2. Метрики регресії: Використовуються для оцінки якості моделей у задачах регресії, де потрібно передбачити неперервну цільову змінну. До популярних метрик регресії належать середня квадратична помилка (MSE), середня абсолютна помилка (MAE), коефіцієнт детермінації (R^2) та середньоквадратична помилка відносно середнього ($RMSE$).
3. Метрики кластеризації: Використовуються для оцінки якості групування прикладів у задачах кластеризації. Деякі звичні метрики кластеризації включають коефіцієнт силуета (*silhouette coefficient*), індекс Данна (*Dunn index*), коефіцієнт Ренді (*Rand index*) та інші.

Аналіз метрик дозволяє отримати уявлення про те, наскільки добре модель працює для конкретної задачі. Важливо розуміти, які метрики важливі для конкретної ситуації та які фактори впливають на їхні значення. Наприклад, метрика може бути чутлива до дисбалансу класів, шуму або викидів у даних. Аналіз метрик допомагає виявити слабкі сторони моделі

Рис. 2.7. Влучність (precision), Повноту (recall) [18]



та покращити її результати шляхом виправлення проблем.

Перед тим, як перейти до розгляду метрик, що використовуються для оцінки результатів класифікації, варто розібратися з ключовою концепцією, що допомагає описати ці метрики - матрицею помилок (confusion matrix).

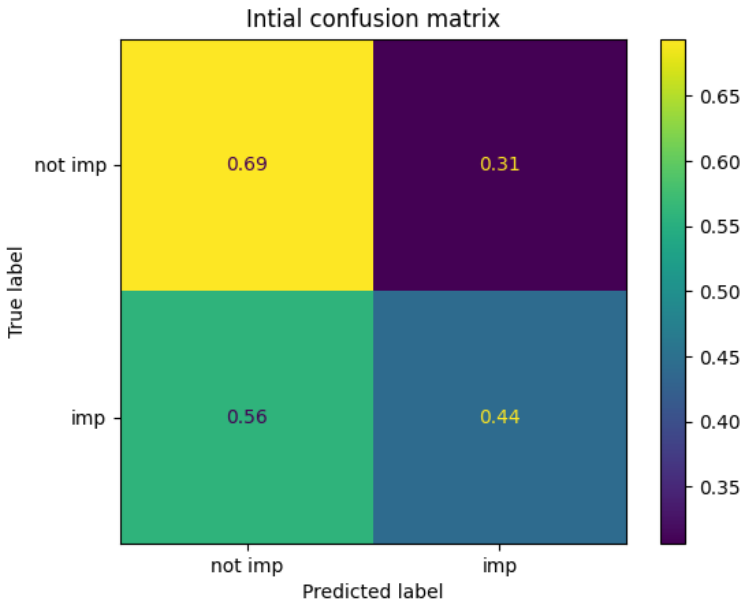
Матриця помилок є потужним інструментом для аналізу результатів класифікації. Вона дозволяє систематично подивитися на прогнози моделі та фактичні значення класів, що допомагає оцінити її точність та недоліки.

Матриця помилок можна уявити у вигляді таблиці 2.8, де по вертикалі відображені фактичні класи, а по горизонталі - прогнозовані класи. Кожна клітинка матриці представляє кількість прикладів, що належать до відповідного поєднання фактичного та прогнозованого класів.

Матриця помилок складається з чотирьох основних елементів:

— True Positive (TP) - кількість прикладів, які правильно класифіко-

Рис. 2.8. Матриця помилок



вані як позитивні.

— True Negative (TN) - кількість прикладів, які правильно класифіковані як негативні.

— False Positive (FP) - кількість прикладів, які помилково класифіковані як позитивні (тип I помилка).

— False Negative (FN) - кількість прикладів, які помилково класифіковані як негативні (тип II помилка).

Матриця помилок надає важливу інформацію про роботу класифікаційної моделі. На основі значень TP, TN, FP та FN можна обчислити різні метрики, такі як точність, повнота, специфічність та F-міра, які використовуються для оцінки якості моделі та розуміння її роботи в різних аспектах. Наше дослідження процює з задачею класифікації, тому розгляниемо метрики, які будуть застосовуватися для оцінки результатів моделей.

В нашому досліджені ми маємо задачу класифікації, тому варто розглянити наступні метрики, які можуть будути застосовані для оцінки результатів моделей.

2.13.1. Точність моделі.

Точність (accuracy) - це дуже інтуїтивно зрозуміла та поширена метрика, яка використовується для оцінки точності класифікації моделі. Вона визначає, яка частка прикладів була правильно класифікована. Точність розраховується за допомогою наступної формули:

$$\text{Accuracy} = (TP + TN)/(TP + TN + FP + FN)$$

Хоча точність (accuracy) є зручною метрикою для швидкого оцінювання точності моделі, вона має обмеження у випадках, коли класи у наборі даних нерівномірні. Це означає, що якщо в наборі даних переважає один клас, то модель може досягти високої значення accuracy, навіть якщо вона має низьку ефективність у класифікації менш представлених класів. Наприклад, якщо в тренувальному наборі даних 90% прикладів належать до класу A, а лише 10% - до класу B, модель, яка завжди передбачає клас A, буде мати високе значення accuracy, але вона може бути некоректною для передбачення класу B.

Для таких випадків більш інформативними можуть бути інші метрики, такі як влучність (precision), повноту (recall), F-міра (F-score), які враховують баланс між класами та забезпечують загальнішу оцінку ефективності моделі.

2.13.2. Влучність та повнота.

Влучність (precision) - це метрика, яка вимірює точність класифікації позитивних прикладів моделі. Вона визначає співвідношення правильно класифікованих позитивних прикладів (TP) до всіх прикладів, які модель визначила як позитивні (TP + FP). Це дає нам інформацію про точність моделі при визначенні позитивних класів.

$$Precision = TP/(TP + FP)$$

Повнота (recall)- це метрика, яка вимірює здатність моделі знаходити

всі позитивні приклади. Вона визначає співвідношення правильно класифікованих позитивних прикладів (TP) до загальної кількості позитивних прикладів у наборі даних (TP + FN). Це дозволяє нам оцінити, наскільки ефективно модель виявляє позитивні приклади.

$$Recall = TP/(TP + FN)$$

Одна з важливих переваг Влучності та повноти (Precision та Recall) полягає в тому, що вони не залежать від відношення класів, на відміну від точності (accuracy). Тому, коли маємо справу з незбалансованою вибіркою, де один клас переважає над іншим, влучність та повнота** є більш інформативними метриками для оцінки ефективності моделі та виявлення проблем у класифікації менш представлених класів.

2.13.3. F1-міра (F-1 Score).

Існує кілька способів комбінування Влучністі та повноти (Precision та Recall) в одну метрику для зведення їх взаємозалежності. Один із таких способів - це використання F1-міри (F1 Score).

F1 Score - це гармонічне середнє влучністі та повноти**, яке дозволяє оцінити баланс між цими двома метриками. Він розраховується за формулою:

$$F1Score = 2 * (Precision * Recall)/(Precision + Recall)$$

Він відображає баланс між влучністю (Precision) та повнотою (Recall). F1 Score може бути в межах від 0 до 1, де значення 1 означає ідеальне поєднання високої влучності та повноти.

Наприклад, у задачах виявлення захворюваннь, ми хочемо мінімізувати як помилкові діагнози (неправильна класифікація здорових людей як хворих), так і пропущені випадки (неправильна класифікація хворих людей як здорових). F1 Score допомагає знайти баланс між цими двома аспектами.

2.13.4. Оцінка F-бета.

F-Beta Оцінка - це розширена версія F1 Score, яка дозволяє враховувати баланс між влучністю і повнотою моделі залежно від ваги, яку ми надаємо одній з цих метрик. Налаштувати вагу можливо за допомогою параметра Beta.

F-Beta Score розраховується за формулою:

$$F{-}BetaScore = (1{+}Beta^2){*}(Precision{*}Recall)/((Beta^2{*}Precision){+}Recall)$$

де Beta - це параметр, який визначає, яку вагу ми надаємо Precision в порівнянні з Recall. Якщо Beta = 1, то отримаємо звичайний F1 Score. Зміна значення Beta дозволяє збалансувати важливість точності і повноти відповідно до потреб конкретної задачі.

Його основна перевага полягає в тому, що він дозволяє налаштовувати баланс між Precision і Recall, залежно від потреб конкретної задачі. Наприклад, якщо важливіше зменшити кількість помилкових позитивних класифікацій, можна використовувати F-Beta Score зі значенням Beta > 1. З іншого боку, якщо більша увага приділяється виявленню всіх позитивних прикладів, можна використовувати F-Beta Score зі значенням Beta < 1.

Ця гнучкість дозволяє налаштовувати F-Beta Score під конкретні потреби задачі класифікації. В результаті, ми отримуємо більш глибоку оцінку ефективності моделі, що враховує бажану збалансованість між Precision і Recall.

2.13.5. Площа під кривою AUC-ROC.

Площа під кривою AUC-ROC (Area Under the Receiver Operating Characteristic Curve) є важливою метрикою, яка використовується для оцінки якості моделей бінарної класифікації. Вона вимірює площу під кривою ROC, яка відображає зв'язок між TPR (True Positive Rate) та FPR (False Positive Rate).

AUC-ROC може мати значення в діапазоні від 0 до 1. Значення 1 відповідає ідеальній моделі, яка максимально відрізняє позитивні та негативні приклади, а значення 0.5 відповідає моделі з випадковою класифікацією.

TPR (True Positive Rate) вимірює відношення правильно класифікованих позитивних прикладів до всіх позитивних прикладів. Формула для розрахунку TPR:

$TPR = TP/(TP + FN)$

FPR (False Positive Rate) вимірює відношення неправильно класифікованих негативних прикладів до всіх негативних прикладів. Формула для розрахунку FPR:

$FPR = FP/(FP + TN)$

Основна перевага AUC-ROC полягає в тому, що вона незалежна від порогового значення і забезпечує комплексну оцінку моделі для всього діапазону можливих значень. Це особливо корисно в задачах з незбалансованими даними або тоді, коли точність і повнота мають різне значення залежно від конкретної ситуації. AUC-ROC дозволяє порівняти ефективність різних моделей та визначити найкращу з них без необхідності встановлювати конкретний поріг класифікації.

РОЗДІЛ 3
ТЕХНІЧНА РЕАЛІЗАЦІЯ

У цій роботі ми досліджуємо оптимальне поєднання візуальних і звукових характеристик, які впливають на стиснення відео. Це дослідження особливо актуальне для відеоконтенту, знятого за допомогою екшн-камер у реальних умовах, де не використовується професійне знімальне обладнання та інструменти постобробки.

Алгоритми машинного навчання, які ми використовуємо, зокрема, збалансований ліс випадкових рішень та метод k-найближчих сусідів (KNN), слугують певній меті. Ці алгоритми дозволяють нам впоратися зі складним завданням обробки широкого спектру ознак у мультимодальних наборах даних. Використовуючи ці алгоритми, ми прагнемо відфільтрувати та ідентифікувати найбільш значущі агреговані ознаки з величезного набору даних. Це сприятиме зменшенню обсягу даних, необхідних для обробки, таким чином оптимізуючи обчислювальні потреби без шкоди для точності наших моделей.

Нижче наведено опис технічної реалізації нашого дослідження з акцентом на наборі даних, який було використано.

3.1. Обробка даних Data processing

Після обрання анотованого мультимодального набору даних, перед нами постала задача розбити його на відповідні частини для тренування та тестування наших моделей. Ми розглядали два потенційні підходи для розділення набору даних.

Перший підхід полягає в розбитті кожного відео в наборі даних на ча-

стини, при якому 80% частин буде використано для тренування, а 20% - для тестування. Такий підхід передбачає, що моделі навчатимуться на 80% кожного відео в наборі даних і тестуватимуться на 20% кожного відео.

У цього підходу є кілька переваг. По-перше, використання більшої частини відео для тренування дозволяє моделям отримати більше інформації та краще враховувати контекстуальні залежності в даних. Крім того, такий підхід може бути ефективним у випадку, коли наш набір даних містить велику кількість відео, оскільки це дозволяє експлуатувати різноманітність даних і покращити узагальнювальну здатність моделей.

Проте, при цьому підході також існують значні недоліки, які можуть викликати проблеми. По-перше, розбиття відео на окремі частини може призвести до втрати контексту та залежностей між фрагментами відео. Це особливо критично в сценаріях, де залежності в часі мають важливе значення для розуміння даних. Крім того, якщо набір даних має обмежену кількість відео, таке розбиття може призвести до недостатнього обсягу даних для ефективного навчання моделей і може виникнути проблема недонавчання.

Другий підхід полягає в розбитті всього набору даних на частини, де 80% від усіх відео використовується для тренування, а 20% - для тестування використованих моделей. Цей підхід має значну перевагу порівняно з першим підходом у тому, що він забезпечує використання більш широкого спектру даних для тестування. Таке розбиття дозволяє оцінити продуктивність моделей на різних видових данних, що сприяє кращому розумінню загальної здатності моделей на різних типах відео.

Для цілей нашого дослідження найбільш логічним підходом буде другий оскільки він гарантує більшу впевненість, що наша модель не буде контекстно залежна і буде мати більш точні результати на даних, які вона до цього не бачила.

Оскільки набір даних є анотованим, у нас є можливість редагувати па-

раметри порогу важливості. Ми можемо змінювати параметр від 1 до 5, щоб визначити, який саме поріг важливості повинен мати фрагмент відео, щоб вважатися "важливим". Можна вважати, що якщо поріг важливості встановлений на 2, це означає, що хоча б двоє анотаторів з 5 анотували їх як важливі. Таким чином, ми можемо змінювати параметри порогу важливості, що дозволить вибирати, яка саме кількість даних в нашому наборі є важливою. Цей інструмент грає дуже важливу роль під час імплементації керованої ансамблевої моделі Balanced Random Decision Forest та керованої непараметричної моделі KNN.

Наступним етапом нашого дослідження є нормалізація даних. Для досягнення цієї мети ми використовуємо метод MinMaxScaler, який дозволяє нормалізувати діапазон значень, зберігаючи при цьому кореляцію між ними. Цей процес є важливим у випадку, коли ми працюємо з відео- та аудіоознаками, оскільки ці дані можуть мати різні шкали та діапазони.

Якщо дані не нормовані, то можуть виникнути наступні проблеми: По-перше, ненормалізовані дані можуть мати різні діапазони значень, що ускладнює порівняння та аналіз цих даних. Наприклад, відеоознаки можуть мати значення від 0 до 255, тоді як аудіоознаки можуть мати значення від -1 до 1. Без нормалізації цих значень, моделі можуть неправильно оцінювати важливість окремих ознак і допускати помилки в результатах.

По-друге, ненормалізовані дані можуть впливати на збіжність та швидкість навчання моделей. Якщо маємо ознаки з великими числовими значеннями порівняно з іншими ознаками, це може призводити до незбалансованості ваг в моделі. Це може спричинити труднощі в оптимізації та зниження швидкості навчання моделі.

Нормалізація даних за допомогою MinMaxScaler дозволяє уникнути цих проблем. Вона перетворює значення в межі від 0 до 1, зберігаючи при цьому кореляцію між ознаками. Це допомагає забезпечити більш рівномірну вагу ознак у моделі та покращує збіжність під час навчання.

Після процесу нормалізації даних, наступним етапом нам потрібно здійснити згладжування даних, щоб усунути прогалини в наборі даних, забезпечивши більшу цілісність даних та зменшивши обривчастість результату. Для здійснення процесу згладжування даних ми використовуємо медіанний фільтр, для якого ми можемо задавати параметр довжини пошукового вікна.

Медіанний фільтр було використано у двох етапах. Початкова стадія включає в себе застосування медіанного фільтра з довжиною пошукового вікна, що дорівнює трьом. В ході цієї процедури, фільтр працює над усім масивом даних, виконуючи згладжування всіх виявлених непослідовностей. Для ілюстрації, якщо в нашому наборі даних є три односекундних кадри, які йдуть один за одним, і два кадри по боках анотовані як "інформативні а середній між ними кадр не має анотації "інформативний то в такому випадку медіанний фільтр зможе згладити подібні прогалини.

Після цього, ми використовуємо ще один медіанний фільтр, але вже з довжиною вікна, що дорівнює п'яти, що дозволяє нам видалити занадто короткі "інформативні"фрагменти. Їх включення в фінальний стиснутий відеоряд не буде продуктивним, оскільки з такого короткого фрагменту важко зрозуміти вкладений в нього контекст. Таким чином, процедурно, пошукове вікно видаляє всі "інформативні"фрагменти відео, які мають сумарну довжину менше 4 "інформативних"кадрів підряд.

Цей методологічний підхід дозволяє моделі навчатися на більш цілісних та зв'язаних даних, що сприяє підвищенню якості результатів моделі. Більше того, застосування подібного підходу до результатів наших моделей, гарантує "читабельність"та зрозумілість даних.

3.2. Технічна реалізація алгоритмів

У цьому дослідженні ми використовуємо збалансований ліс випадкових рішень та метод k-найближчих сусідів (KNN), щоб впоратися з багатогранною природою наших відеоданих. Ці алгоритми просіюють численні аудіо-візуальні характеристики в наших необроблених відеозаписах з екшн-камер, визначаючи найбільш важливі характеристики для ефективного стиснення відео.

У наступному розділі детально описано конкретне застосування цих алгоритмів, зокрема методу ансамблю, Decision Forest, у нашому дослідженні, а також надано огляд нашого унікального набору даних.

3.2.1. Реалізація ансамблевого методу Decision Forest.

Після проведення глибокого аналізу даних та виявлення, що використовувані нами дані є незбалансованими, було прийнято рішення застосувати модель типу дерева рішень. Ми вибрали ансамблевий метод Balanced Random Decision Forest, який має здатність аналізувати та обробляти незбалансовані дані. Перед початком налаштування моделі, було необхідно визначити, які параметри порогу важливості слід використовувати.

Було встановлено поріг важливості, що дорівнює 1. Це обумовлено тим, що при встановленні порогу важливості на рівні 2, лише 5% від загальної кількості даних будуть вважатися важливими, що є недостатньою величиною для ефективного навчання моделі. Таким чином, кількість важливих моментів у відео становить приблизно 20%, і модель отримує значно більше позитивних прикладів для навчання.

Для оцінки результатів моделі ми вирішили використовувати F1-міра (F-score), як основний показник метрики, замість показника точності (accuracy). Оскільки наші дані є вкрай незбалансованими, точність в таких умовах не є достатньо точним критерієм оцінювання.

Для визначення найкращих параметрів для Balanced Random Decision Forest ми використовували метод Grid-Search (пошук по сітці) з орієнтацією на покращення показника F1-міри. Таким чином, ми можемо бути впевнені, що обійшли всі вузли сітки, знайшовши оптимальні параметри для більш широкого кола даних, включаючи всі важливі моменти, що раніше могли бути не враховані при оцінці оптимальних параметрів моделі.

3.2.2. Реалізація алгоритму k-найближчих сусідів (KNN). Зважаючи на те, що оцінка важливості параметрів може залежати від моделі, ми вирішили паралельно з Decision Forest використовувати іншу модель, яка навчалася без вчителя, щоб отримати більш різноманітні результати та надати альтернативні погляди на проблему. Для цього ми обрали керовану непараметричну модель KNN. Ця модель була обрана насамперед через простоту реалізації та структурну прозорість, що дозволяє більш детально зрозуміти висновки моделі та фактори, які на них впливають.

Для пошуку оптимальних параметрів для моделі KNN ми використовували ту ж методологію пошуку, що і для Random Balanced Decision Forest. Параметричний пошук показав, що модель дає кращі результати точності та повноти без використання згладжування даних. Згладжування даних може потенційно зашкодити визначенню ключових точок, на основі яких будуються кластери для виявлення найближчих сусідів. Отже, при застосуванні моделі KNN ми змінили наші дані і вирішили відмовитися від процесу згладжування задля досягнення більш точних і повних результатів.

3.2.3. Розробка ознак Feature engineering. Існує дві основні методології для агностичного визначення важливості ознак у використовуваних моделях. Перший - це метод перестановки, заснований на концепції перестановки значень окремих ознак і спостереженні за впливом цього на прогнозний результат моделі. Процес полягає в наступному: якщо ознака

має важливе значення, перестановка її значень у тестовому наборі даних під час прогнозування призводить до значного покращення результатів. І навпаки, якщо ознака вважається неважливою, перестановка її значень не призводить до суттєвої зміни метрики. Отже, метод перестановки полегшує оцінку важливості ознак на основі впливу їхньої перестановки на якість моделі. Наступним методом є метод вилучення стовпців ознак, який використовується для визначення важливості ознак шляхом послідовного видалення кожної ознаки з набору даних і спостереження за зміною продуктивності моделі в результаті. Фундаментальною передумовою є те, що якщо видалення ознаки призводить до значної зміни метрики якості моделі, то ця ознака вважається важливою. Однак метод вилучення стовпців ознак відрізняється від методу перестановок. Він дозволяє оцінити важливість ознаки без необхідності перенавчання моделі для кожної перестановки, що робить його більш ефективним з точки зору обчислень. Однак метод вилучення стовпців ознак може не виявити важливість взаємодії ознак, оскільки він розглядає кожну ознаку незалежно. Хоча метод перестановок може бути більш точним у визначенні важливості ознак, він вимагає більше обчислювального часу.

Після застосування обох методів (перестановки та вилучення стовпців) для керованої непараметричної Random Balanced Decision Forest моделі було виявлено, що ці методи не дає нової інформації. Коли видаляється стовпець, результати моделі залишаються незмінними, оскільки модель просто замінює його іншим стовпцем, щоб отримати ту саму інформацію. У випадках, коли точність моделі змінюється після вилучення деяких стовпців, незалежно від того, який саме стовпець вилучено, точність змінюється на однакову кількість значень. Це свідчить про значний ступінь кореляції. Це також було підтверджено дендрограмою, яка виявила численні низькорівневі зв'язки коефіцієнтів ознак, що вказувало на необхідність розділення даних. За умови колінеарності ознак заміна однієї ознаки не матиме зна-

чного впливу на результат моделі, оскільки вона може отримати ту ж саму інформацію з корельованої ознаки. Для цього ми використали коефіцієнт кореляції Спірмена для визначення демаркаційної лінії для поділу на кластери. Ієрархічна кластеризація на основі рангових кореляцій Спірмена зумовлює вибір порогового значення та збереження однієї ознаки для кожного кластера. Будь-які дані, що демонстрували кореляцію, нижче цієї лінії, було поділено на кластери, де, усі, крім одної ознаки, були виключені, що призвело до утворення поділення ознак на нові менш корельовані кластери нння цього порогу призвело до поділу даних на задовільні для нас кластери 3.1. Процес вибору порогу включав приблизно 100 ітерацій з рі-

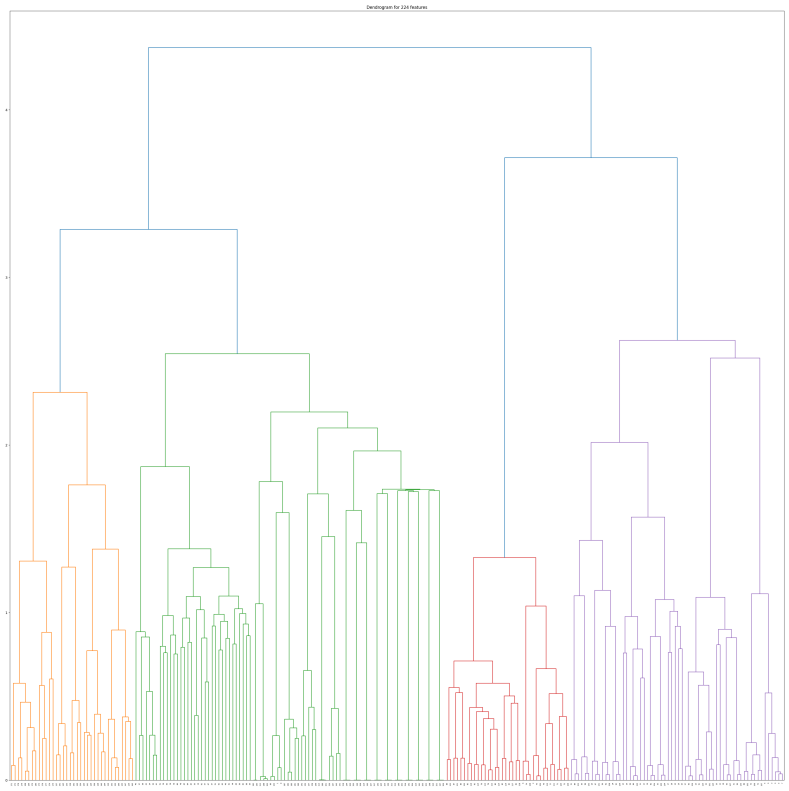


Рис. 3.1. Дендрограма до кластеризації на основі рангових кореляцій Спірмена

зними параметрами порогу. Наш підхід був ітеративним: ми встановлювали поріг, відповідно розбивали дані, запускали модель і оцінювали результати на предмет їхньої задовільності. F1-міра була нашою основною оціночною метрикою. Після проведення цих тестів ми дійшли висновку, що поріг 0,24 дає найвищий показник F1-міри.

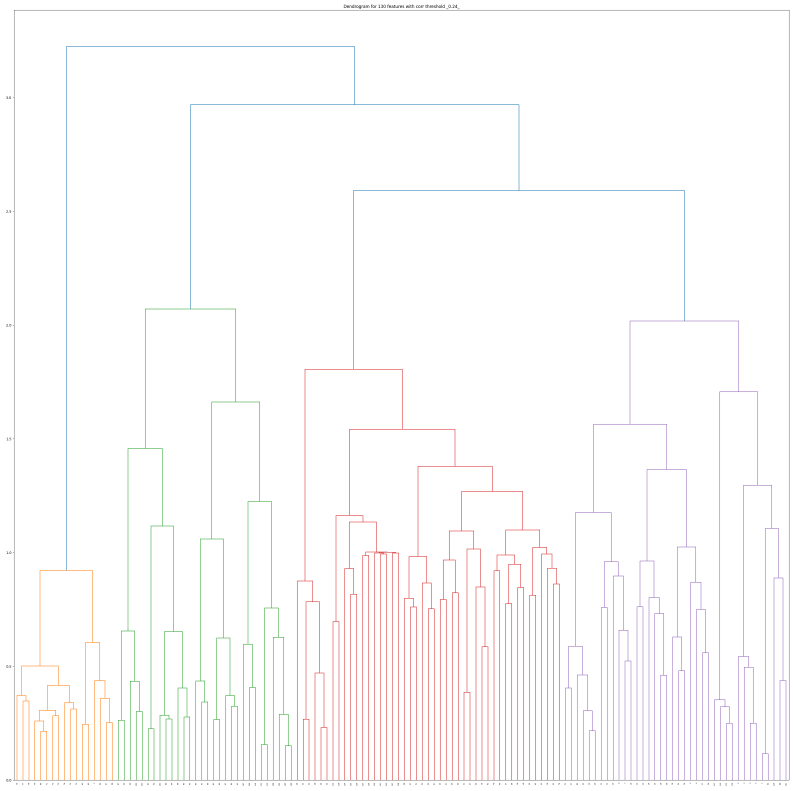


Рис. 3.2. Дендрограма після виключення корелюючих ознак з коефіцієнтом кореляції 0.24

Застосування цього порогу призвело до поділу даних на задовільні для нас кластери 3.2. З кожного кластера ми обрали одну ознаку, виключивши всі інші ознаки, які показали кореляцію з обраною ознакою. Такий підхід забезпечив збереження унікальної інформації та потенційної значущості кожного кластера, уникнувши при цьому повторення даних через кореляцію.

Після цього процесу у нас залишилося 130 ознак. Ми застосували метод вилучення стовпчиків, щоб визначити, які з них є вирішальними для Decision Tree. Результати не були задовільними, модель все ще показувала ознаки перенавчання. Отже, ми вирішили вилучити більше ознак через коефіцієнт кореляції даних.

Повторно переглянувши дендрограму після першого застосування методу Спірмена, ми помітили чіткий поділ на чотири кластери. Розглянемо ці чотири кластери детальніше для подальшого аналізу.

На попередньому кроці ми використали метод вилучення стовпців показав високий рівень кореляції даних, що утворились, тому все ще призводив до перенавчання Decision Tree, і ми продовжили вилучення ознак.

В кожному з чотирьох кластерів будуємо розбиття по групах корельованих між собою ознак. Використовуючи метод вилучення стовпців, оцінюємо зміну метрик з та без вилученої ознаки. Вилучення ознаки може як позитивно так і негативно вплинути на якість моделі. Позитивний вплив відображається додатними або від'ємними значеннями в таблиці. Що меншим число є в таблиці тим точнішою буде модель за вилучення цієї ознаки. Для кожної групи розбиття, ми обрали ознаку, яка має найменший вплив на зміну результату, тобто є найважливішою з кластера. Ця ознака може є найбільшим від'ємне число, всі інші зібрати в окремий список, список ознак для можливого вилучення. Варто зазначити, що ознаки які позитивно впливають, що відповідають додатнім значенням в таблиці ми не вилучаємо. Позитивна різниця в точності вказувала на те, що модель буде працювати гірше без цієї ознаки. Під час виконання це виглядало, як таблиця, структурована за спаданням, що відображала різницю в точності моделі з вилученням ознак і без нього.

Наступним кроком було ітерування по цьому списку, перевіряючи методом викреслення стовпців доцільність вилучення даної ознаки, якщо так - вилучаємо та продовжуємо ітерувати без неї, ні - залишаємо.

Утворений список ознак, які ми залишаємо склав 121 ознаку (Додаток 1).

У випадку методу k-найближчих сусідів (KNN), після першої ітерації кореляцій Спірмена, відбір ознак на другій ітерації не був доцільним, оскільки він не дав жодних результатів. Тому у випадку методу KNN ми залишили 130 ознак як найбільш ефективні (Додаток 2).

Наступним етапом ми використовуємо бібліотеку SHAP (Shapley Additive Explanations) для з'ясування важливості окремих ознак у машинному навчанні. SHAP ґрунтується на теорії ігор і концепції значень Shapleyi, що дозволяє розподілити внесок кожної ознаки в прогноз моделі серед усіх можливих комбінацій ознак. Ця методика дає детальне уявлення про вплив кожної ознаки на прогнози моделі.

Бібліотека SHAP дозволяє нам визначити важливість окремих ознак шляхом обчислення значень SHAP для кожного екземпляра даних. Це дає нам можливість зрозуміти, які ознаки найбільш суттєво впливають на вихідні прогнози моделі, а також напрямок цього впливу.

Результати нашої роботи з бібліотекою SHAP будуть представлені в розділі 4.3 Важливість Ознак (Feature Importance), де ми проаналізуємо важливість окремих ознак у нашій моделі та з'ясуємо їхній вплив на кінцеві прогнози.

РОЗДІЛ 4
РЕЗУЛЬТАТИ

4.1. Результати метрик моделей

Варто зазначити, що в цьому розділі ми порівнюємо результати моделей навчених на оригінальному наборі данних, без застосування технік відбору ознак.

Для задачі підсумування відео з використанням мультимодального набору даних ми провели дослідження на керованій непараметричній моделі Random Balanced Decision Forest з пошуком гіперпараметрів націлених на максимізацію F1-міри (F1-Score).

Найкраще себе проявила модель з наступними параметрами:

— 'criterion': 'entropy',

— '$max\_features$': 'sqrt'

— '$n\_estimators$': 100

— '$sampling\_strategy$': 'not minority'

Для отримання наступних результатів ми також використали зглажування даних з довжиною пошукового вікна $'med\_thres' : 5,' hard\_thres' : 3$. При цих параметрах нам вдалося досягти найкращих результатів моделі.

Нижче наведені результати моделі Random Balanced Decision Forest, налаштованої на максимізацію F1-міри:

Метрика	Результат
Accuracy	0.63215
F1-Score	0.52214
AUC - ROC	0.62593

Нам також вдалося знайти гіперпараметри для моделі, які максимізували показник Accuracy:

— 'criterion': 'entropy',

— '$max\_features$': 'sqrt'

— '$n\_estimators$': 100

— '$sampling\_strategy$': 'not majority'

З використанням зглажування даних з параметрами: $'med\_thres' : 5,' hard\_thres'$
При використанні цих параметрів ми отримали наступні метрики моделі:

Метрика	Результат
Accuracy	0.69689
F1-Score	0.29524
AUC - ROC	0.56921

Для наступної частини роботи був використаний метод k-найближчих сусідів (k-nearest neighbor method, KNN), для якого ми знайшли гіперпараметри, націлені на максимізацію F1-міри:

— 'metric': 'minkowski',

— '$n\_neighbors'$: '2',

— '$weights'$: 'distance'

При використанні цих параметрів ми отримали наступні результати метрик моделі:

Метрика	Результат
Accuracy	0.61052
F1-Score	0.42867
AUC - ROC	0.56783

Аналогічно, ми спробували знайти найкращі гіперпараметри для KNN моделі, які націлені на максимізацію точності (Accuracy). Для цього використовувалися наступні гіперпараметри:

— 'metric': 'manhattan',

— '$n\_neighbors$': 42,

— 'weights': 'uniform'.

При використанні цих гіперпараметрів для KNN моделі, націлених на максимізацію точності (Accuracy), ми отримали наступні результати:

Метрика	Результат
Accuracy	0.66721
F1-Score	0.38271
AUC - ROC	0.57737

Варто зазначити, що після процесу відкидання корельованих ознак, на наступному кроці, значення метрик моделі покращилися в середньому на 10%.

4.2. Відбір ознак

У нашому дослідженні ми застосували кілька методологій для оцінки значущості ознак. Використовуючи коефіцієнт кореляції Спірмена, ми визначили поріг 0,24 для кластеризації, що призвело до поділу даних на задовільні кластери. З кожного кластера ми вибрали по одній ознаці, виключивши інші, які показали кореляцію, в результаті чого у нас залишилося 130 ознак.

Потім ми застосували метод вилучення стовпців, щоб визначити найважливіші ознаки для дерева рішень. Однак результати виявилися незадовільними, що змусило нас проробити алгоритм ще раз. Повторно переглянувши дендрограму, ми помітили чіткий поділ на чотири кластери. Ми побудували розбиття за групами корельованих ознак всередині цих кластерів і оцінили зміну метрик з вилученою ознакою та без неї.

Після ітерацій по цьому списку та перевірки доцільності видалення ознаки, ми залишили 121 ознаку, список дивитись Додаток 1. У випадку

методу k-найближчих сусідів (KNN) після першої ітерації кореляцій Спірмена не було доцільно відбирати ознаки на другій ітерації, оскільки вона не дала жодних результатів. Тому у випадку з методом KNN ми залишили 130 ознак, список дивитись Додаток 2, як найбільш ефективні.

На наступному етапі дослідження ми використаємо бібліотеку SHAP для визначення важливості окремих ознак у машинному навчанні.

4.3. Важливість ознак

Використовуючи метод SHAP (SHapley Additive exPlanations), поглянемо на візуалізації важливості ознак:

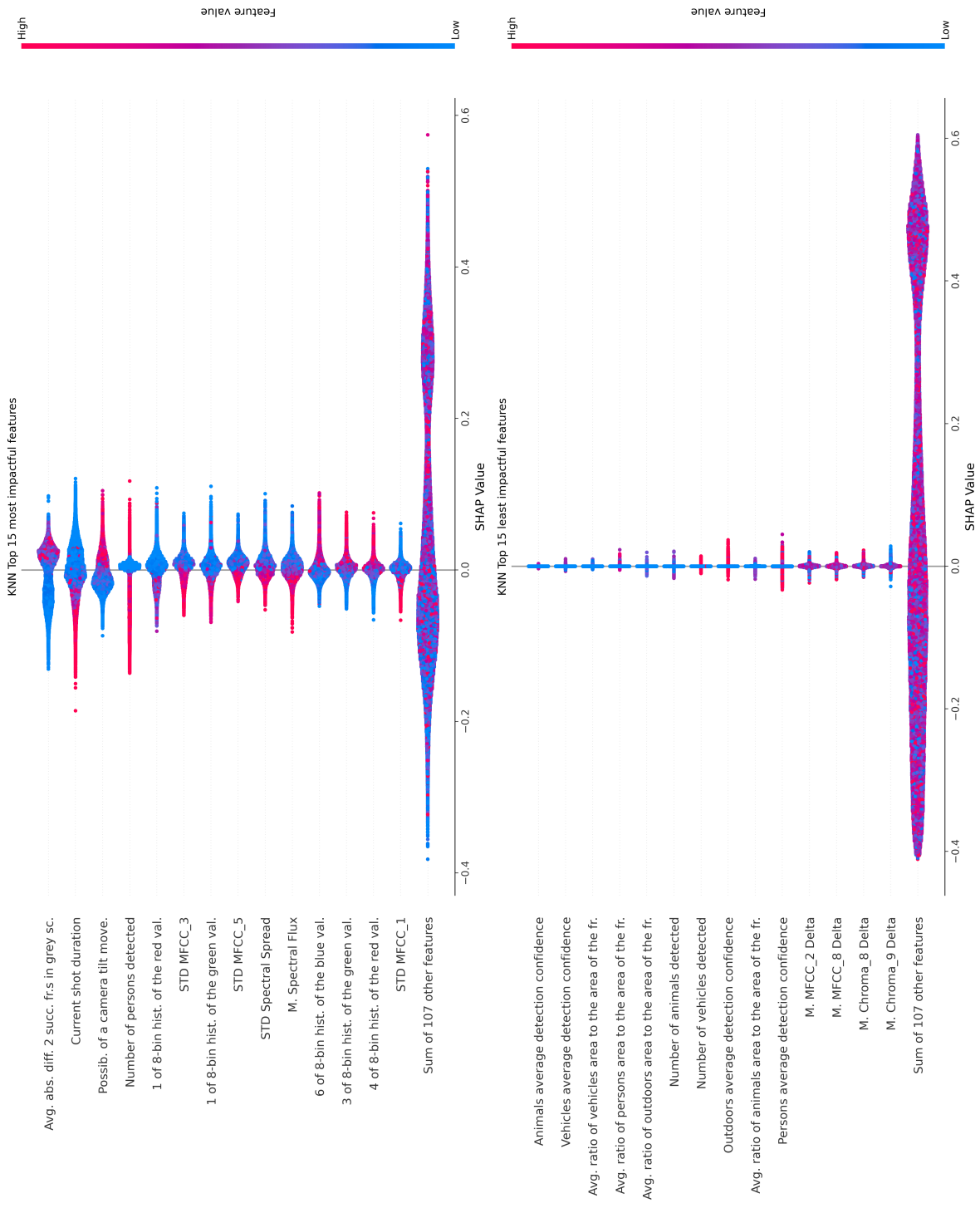


Рис. 4.1. KNN Топ 15 найбільш/найменш корисних ознак

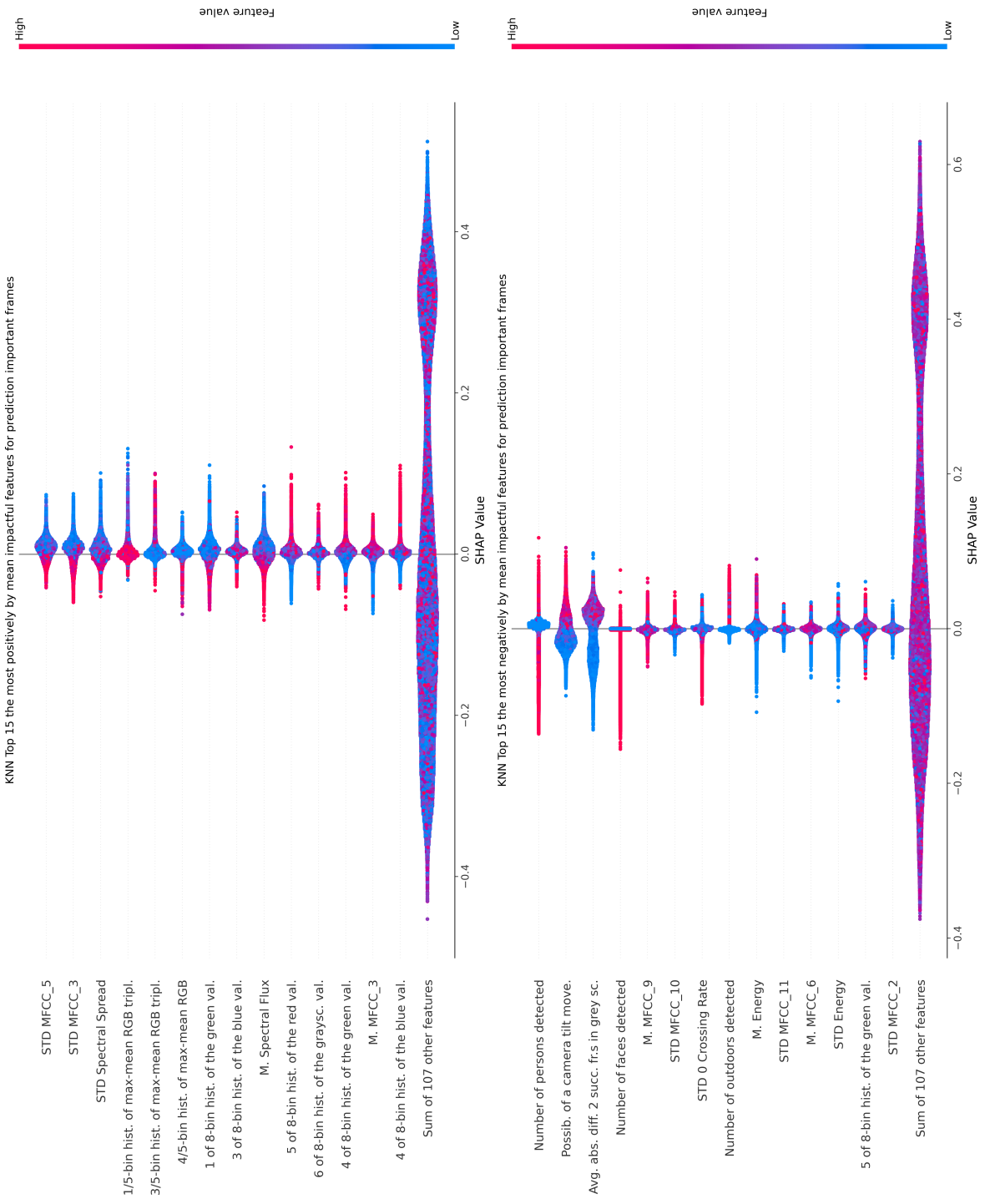


Рис. 4.2. KNN Топ 15 зміщених позитивно/негативно ознак

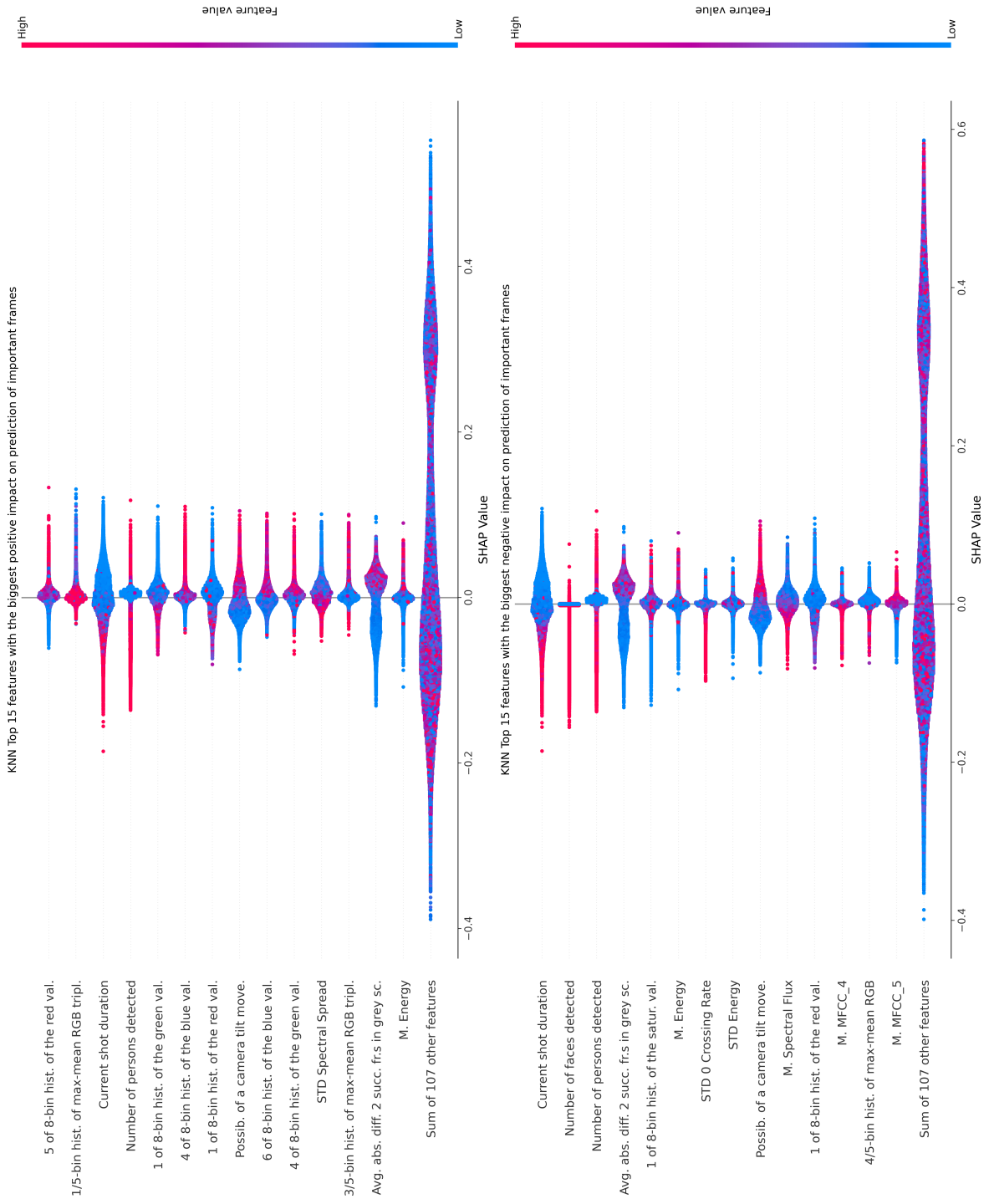


Рис. 4.3. KNN Топ 15 ознаки, які внесли найбільший позитивний вклад в окремому взятих випадках

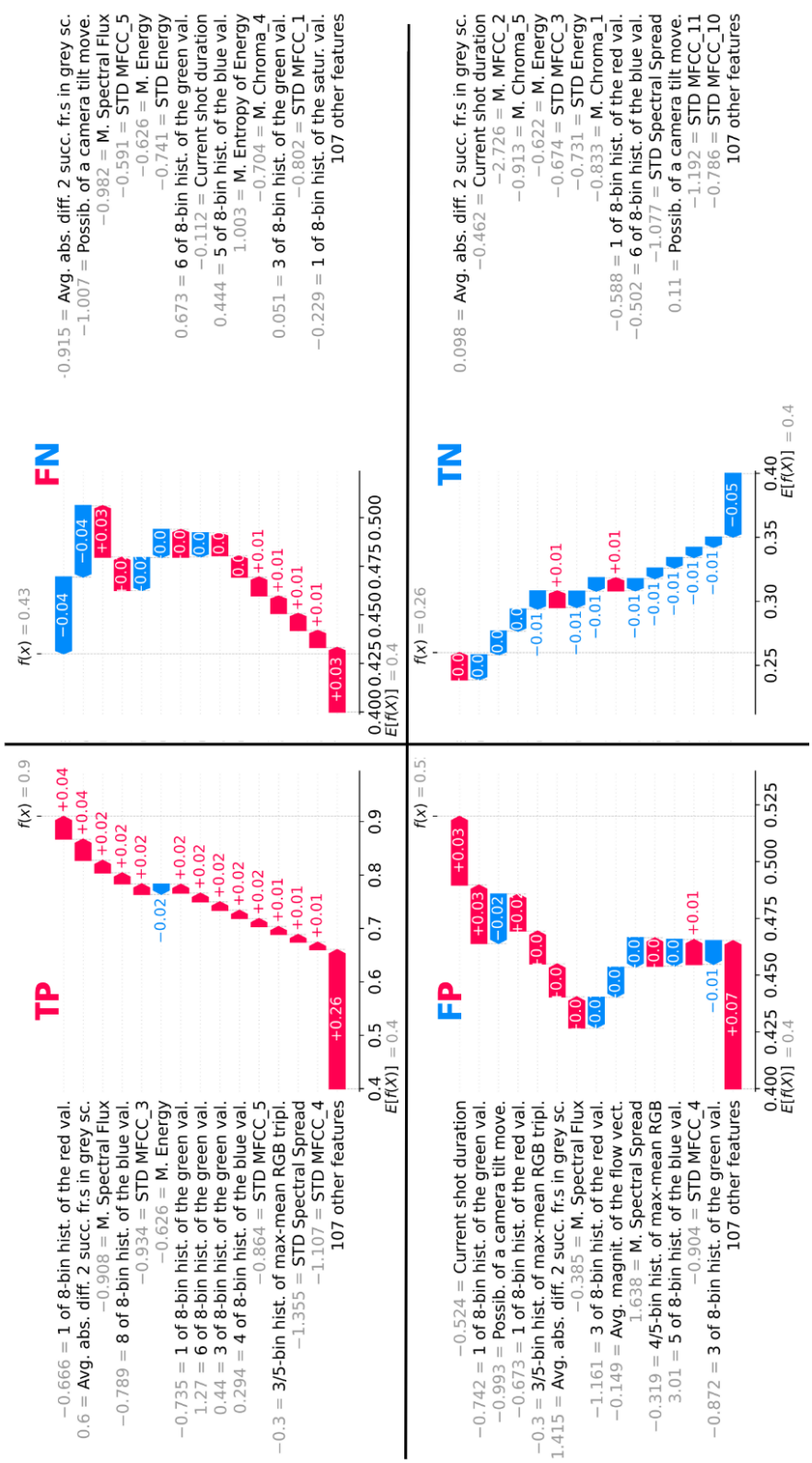


Рис. 4.4. KNN Візуалізація впливу окремих ознак як матриці невідповідностей

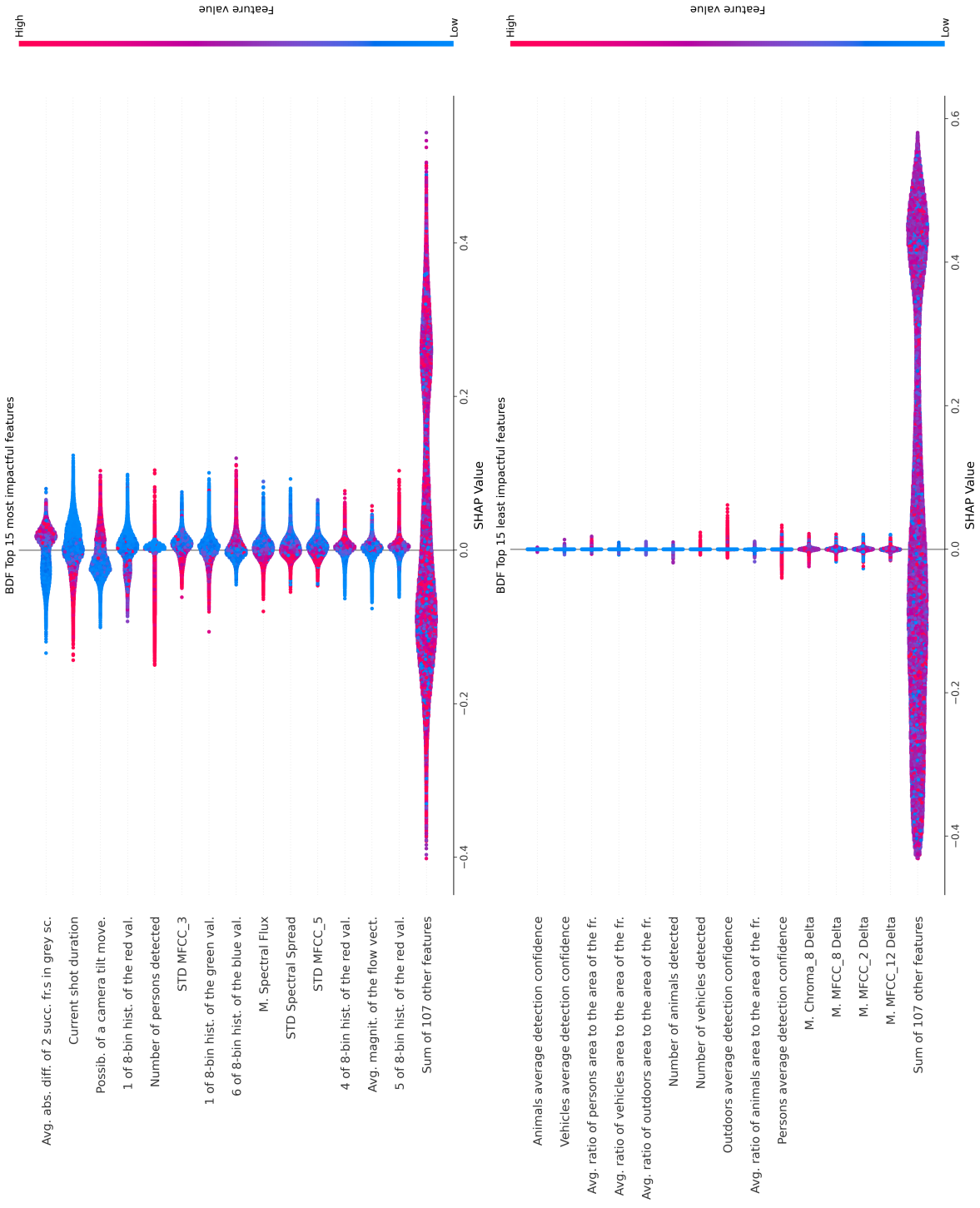


Рис. 4.5. BRF Топ 15 найбільш/найменш корисих ознак

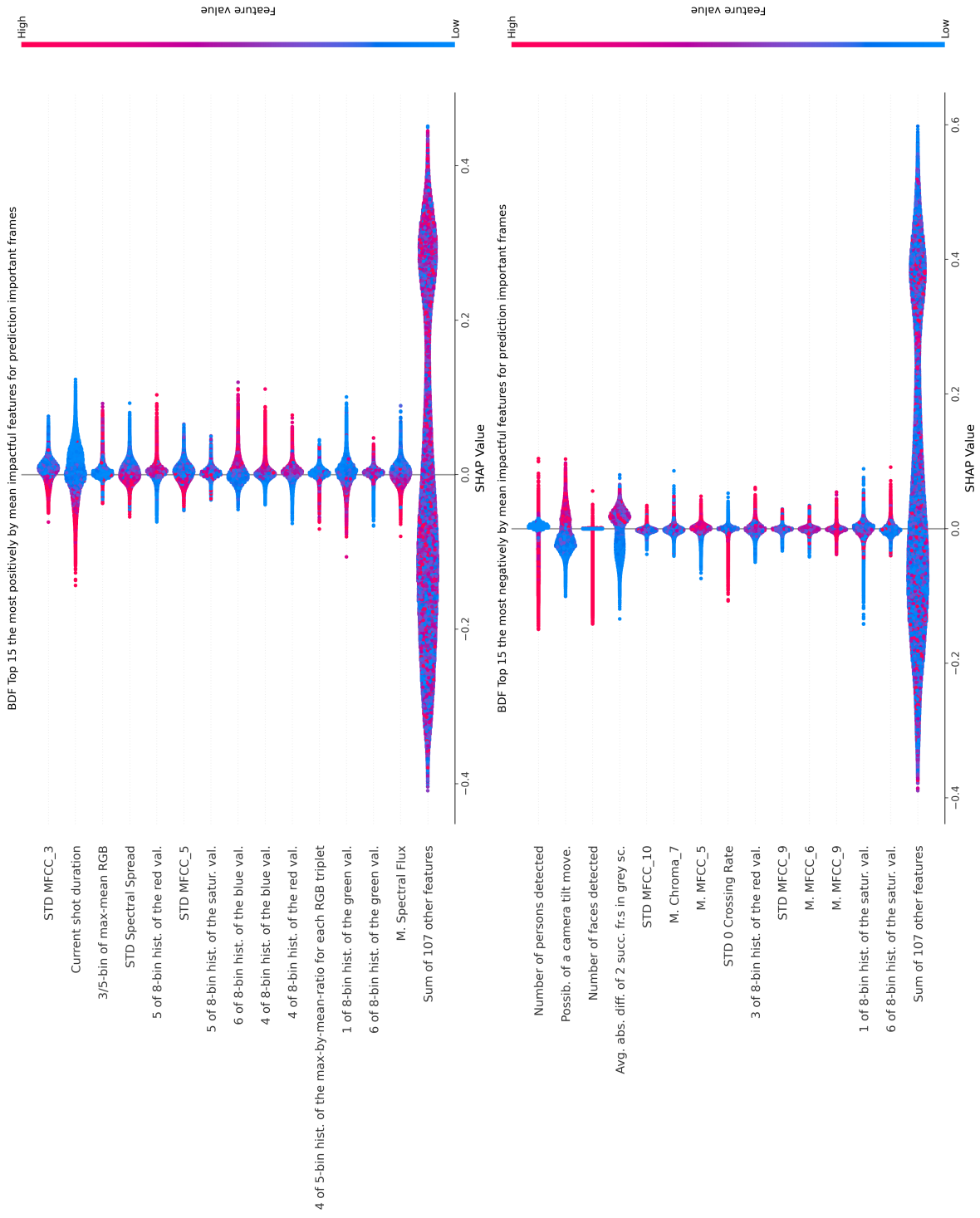


Рис. 4.6. BRF Топ 15 зміщених позитивно/негативно ознак

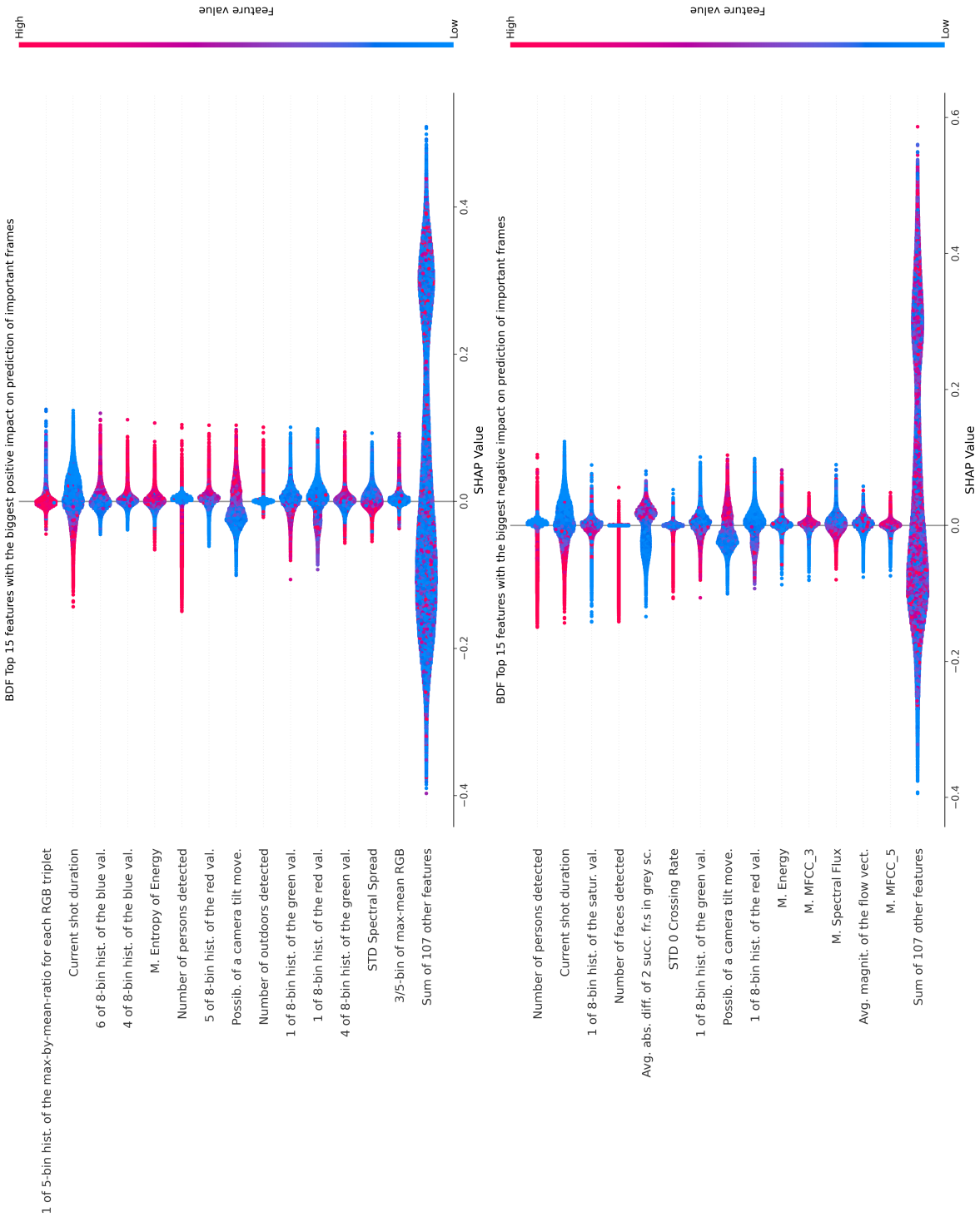


Рис. 4.7. BRF Топ 15 ознаки, які внесли найбільший позитивний вклад в окремому взятих випадках

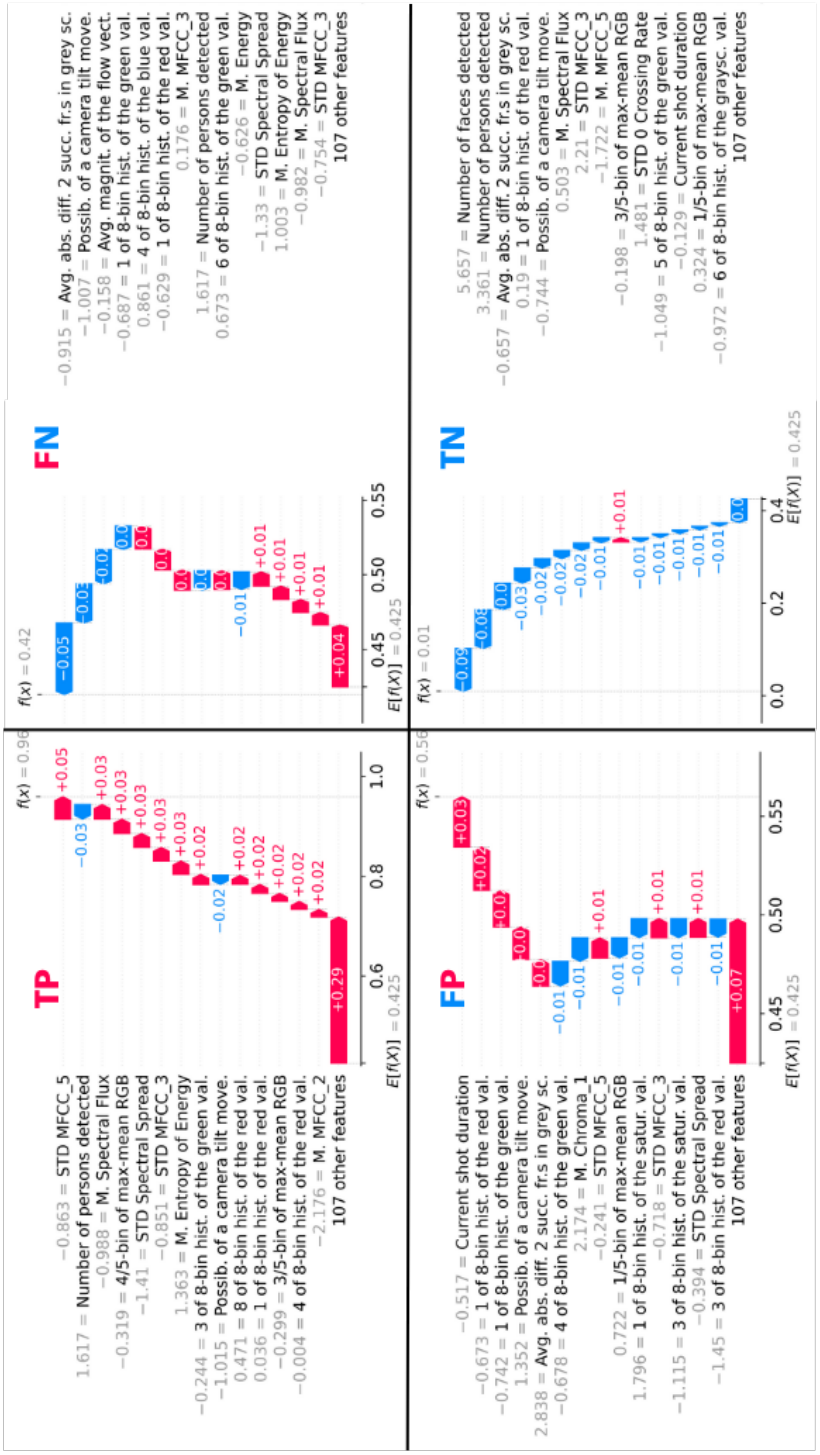


Рис. 4.8. BRF Візуалізація впливу окремих ознак як матриці невідповідностей

4.4. Обговорення результатів

Таблиця 4.1

Ознаки з найбільш загальним вкладом

Назва ознаки	**KNN**	**BRF**
Середня абсолютна різниця між двома послідовними кадрами в градаціях сірого	✓	✓
Довжина зйомки кадру	✓	✓
Можливість зміни нахилу камери	✓	✓
1-8 біт гістограми червоних значень	✓	✓
Кількість виявлених людей	✓	✓
STD MFCC 3	✓	✓

Було виявлено, що для обох моделей найбільший внесок мають такі характеристики:

— **Середня абсолютна різниця двох послідовних кадрів у відтінках сірого**: Ця ознака вимірює зміну інтенсивності між двома послідовними кадрами, що може вказувати на значні зміни сцени або точки дії у відео. Більша різниця означає більш динамічний вміст, що може вважатися більш важливим для стиснення.

— **Тривалість поточного кадру**: Тривалість поточного кадру часто відображає його важливість. Довші кадри, як правило, містять більше інформації або ключових подій, які мають вирішальне значення для розуміння відеоконтенту, що робить цю характеристику ефективним індикатором важливих сегментів.

— **Можливість нахилу камери**: Вимірює ймовірність нахилу камери в поточному кадрі. Рухи камери, такі як нахил, часто слугують для того, щоб підкреслити певний об'єкт або дію у відео, таким чином підвищуючи його значущість для стиснення.

— **1 з 8-бітової гістограми значення червоного кольору**: Ця функція відображає розподіл інтенсивності червоного кольору в поточному кадрі. Оскільки червоний колір часто може відповідати значущим об'єктам або сценам у відео, ця функція може бути корисною для ідентифікації важливих сегментів.

— **Кількість виявлених осіб**: Присутність людей часто вказує на ключові моменти або взаємодії у відео. Отже, кількість людей, виявлених у кадрі, може бути сильним показником його релевантності.

— **Середньоквадратичне відхилення центральних коефіцієнтів Mel-частоти ($MFCC_3$)**: Ця функція фіксує варіацію третього $MFCC$, який представляє спектральні властивості аудіодоріжки у відео. Висока варіабельність може бути пов'язана зі значними звуковими подіями, що підвищує важливість кадру.

Разом з тим, в обох моделях були виявлені ознаки з найменшим внеском (Таблиця 4.2):

— **Достовірність виявлення та середнє співвідношення площ для тварин, транспортних засобів та вуличних територій до розміру кадру**: Ці ознаки були менш релевантними для стиснення відео, ймовірно, тому, що наявність або співвідношення цих елементів не обов'язково корелює з ключовими моментами відео.

— **Кількість виявлених тварин і транспортних засобів**: Як і в попередньому випадку, кількість виявлених об'єктів не зробила значного внеску в процес стиснення, ймовірно, через їхню низьку кореляцію з ключовими моментами відео.

— **Середня достовірність виявлення осіб**: Як не дивно, хоча кількість виявлених осіб була важливою, впевненість виявлення осіб не зробила значного внеску в моделі. Це може свідчити про те, що сама присутність людей, незалежно від достовірності виявлення, є більш важливою для ідентифікації ключових відеосегментів.

Таблиця 4.2

Ознаки з найменшим загальним вкладом

Назва ознаки	**KNN**	**BRF**
Середня достовірність виявлення тварин	✓	✓
Середня достовірність виявлення транспортних засобів	✓	✓
Середнє співвідношення людей до площі кадру	✓	✓
Середнє співвідношення площі кадру до площі транспортних засобів	✓	✓
Середнє співвідношення площі кадру до площі відкритого простору	✓	✓
Кількість виявлених тварин	✓	✓
Кількість виявлених транспортних засобів	✓	✓
Середня достовірність виявлення вуличних територій	✓	✓
Середнє співвідношення тварин до площі кадру	✓	✓
Середня достовірність виявлення людей	✓	✓

Під час дослідження ознак, які позитивно впливали на моделі для стиснення відео, ми помітили, що певні ознаки мали вагомий точковий вплив.

Для обох розглянутих моделей найбільший позитивний точковий вплив мали такі ознаки:

Таблиця 4.3

Ознаки з найбільшим позитивним вкладом в окремих взятих випадках

Назва ознаки	**KNN**	**BRF**
1-5 біт гістограма максимального середньоквадратичного співвідношення для кожного триплету RGB	✓	✓
Довжина зйомки кадру	✓	✓
Кількість виявлених облич	✓	✓
5-8 біт гістограми червоних значень	✓	✓
1-8 біт гістограми зелених значень	✓	
4-8 біт гістограми синіх значень	✓	✓
1-8 біт гістограми червоних значень	✓	
6-8 біт гістограми синіх значень		✓
Середнє значення ентропії енергії		✓

— **1 з 5-бітової гістограми максимального середньоквадратичного відношення для кожного триплету RGB**: Низькі значення змінювалися від нуля до суцільного позитивного впливу, змішаного з середніми значеннями майже на всьому шляху, тоді як високі значення мали від нуля до злегка негативного впливу. Це свідчить про те, що кадри з низькою або помірною варіацією інтенсивності кольору можуть бути важливими, тоді як кадри з надзвичайно високою варіацією можуть бути менш релевантними для стиснення.

— **Поточна тривалість кадру**: Середні значення мали незначний негативний ефект, низькі значення мали позитивний ефект, а високі значення мали негативний ефект від нуля до значного. Це підкреслює, що надзвичайно довгі або короткі кадри не завжди вказують на важливий зміст.

— **Кількість виявлених осіб**: Низькі значення мали незначний позитивний вплив, тоді як високі значення могли бути як суцільно позитивними, так і суцільно негативними. Це свідчить про те, що важливість присутності людей дуже залежить від контексту.

— **5 з 8-бітової гістограми значення червоного кольору**: Низькі значення змінювалися від незначно позитивних до суцільно негативних, тоді як високі значення були позитивними, від незначно до суцільно позитивних. Це свідчить про те, що певні діапазони інтенсивності червоного можуть бути більш показовими для ключових сегментів відео, ніж інші.

— **4 з 8-бітової гістограми значення синього кольору**: Низькі значення змінювалися від слабко позитивного до суцільно негативного, тоді як високі значення були позитивними, від злегка до суцільно позитивних. Як і для червоного кольору, певні діапазони інтенсивності синього можуть бути більш важливими.

— **1-8 біт гістограми зелених/червоних значень**: Високі та сере-

дні значення мали негативний ефект, тоді як низькі значення змінювалися від незначного негативного до суцільного позитивного. Це підкреслює, що KNN може бути чутливим до певних діапазонів інтенсивності зеленого/червоного кольору, що може допомогти ідентифікувати схожий вміст у відео.

— **6 з 8-бітової гістограми значення синього і 4 з 8-бітової гістограми значення синього**: Низькі значення мали незначний негативний вплив, змішані значення мали незначний позитивний вплив, а високі значення мали сильний позитивний вплив. Це підкреслює здатність BRF використовувати певні діапазони інтенсивності синього для виділення важливих сегментів відео.

— **Середнє значення ентропії енергії**: Високі значення були переважно позитивними з деякими суцільно негативними, середні значення мали нульовий вплив, а низькі значення мали незначний негативний вплив. Це вказує на те, що складність звукової доріжки, виміряна за допомогою ентропії, може бути значною мірою визначальною для ключових подій у відео.

Цікаво, що деякі з ознак, які позитивно вплинули на моделі стиснення відеоматеріалів, у певних випадках також мали значний негативний вплив. Це підкреслює складну природу стиснення відеоматеріалів, де релевантність певних ознак може змінюватися залежно від контексту.

Для обох моделей негативний вплив мали такі характеристики:

— **Кількість виявлених осіб**: Високі значення цієї ознаки мали суцільний негативний або суцільний позитивний вплив, тоді як низькі значення мали незначний позитивний вплив. Це свідчить про те, що наявність великої кількості людей у кадрі не обов'язково означає його важливість. Може статися так, що у відео з великою кількістю людей важливими для стиснення є лише конкретні взаємодії або

Таблиця 4.4

Ознаки з найменшим позитивним вкладом в окремих взятих випадках

Назва ознаки	**KNN**	**BRF**
Кількість виявлених осіб	✓	✓
Довжина зйомки кадру	✓	✓
Кількість виявлених облич	✓	✓
1 з 8-бітової гістограми значення насиченості	✓	✓
Середня абсолютна різниця двох послідовних кадрів у відтінках сірого	✓	✓

події за їхньої участі.

— **Довжина зйомки кадру**: Ця функція показує негативний вплив для високих значень і діапазон від нуля до суцільного позитиву для низьких значень. Це свідчить про те, що довші кадри не завжди є більш інформативними або важливими. Насправді, в деяких випадках коротші кадри можуть містити ключові події або переходи, які мають вирішальне значення для стиснення.

— **1 з 8-бітової гістограми значення насиченості**: Високі значення цієї характеристики мали майже нульовий вплив, змішані з середніми та низькими значеннями, значна частина низьких значень мала негативний вплив. Це свідчить про те, що високо насичені кадри не обов'язково роблять внесок в стиснення. Можливо, такі кадри представляють візуально вражаючий, але в кінцевому підсумку нерелевантний контент, наприклад, переходи або спецефекти.

— **Кількість виявлених облич**: Ця ознака мала майже нульовий вплив для низьких значень, суцільний негативний вплив для висо-

ких значень, а в деяких випадках - незначний позитивний вплив. Це свідчить про те, що наявність облич у кадрі, як і присутність людей, не завжди означає його важливість для резюмування. Лише окремі випадки, можливо, пов'язані з взаємодією або виразом обличчя, можуть бути важливими.

— **Середня абсолютна різниця двох послідовних кадрів у градаціях сірого**: Низькі значення цієї ознаки здебільшого мали негативний вплив, тоді як середні та високі значення мали незначний позитивний ефект. Це свідчить про те, що кадри, які мало відрізняються від своїх попередників, можуть не мати вирішального значення для стиснення. Натомість кадри, які суттєво відрізняються від попередніх, вказуючи на потенційну зміну сцени або важливу подію, можуть бути більш важливими.

Тепер заглибимося в аналіз зміщення системи, досліджуючи окремі особливості та їхні тенденції у впливі на моделі:

Для обох моделей:

— **STD MFCC 3**: Ця ознака має загалом позитивний вплив, збільшуючи важливість кадру приблизно на 5-7%. Це, ймовірно, пов'язано з тим, що Мел-частотні центральні коефіцієнти (MFCC) відображають спектр потужності аудіосигналів, а зміни в цьому спектрі (як показано стандартним відхиленням, STD) можуть вказувати на зміни в аудіоконтексті, які часто асоціюються з важливими подіями у відеоролику. Крім того, коефіцієнти MFCC можуть відображати сприйняття слухової системи людини, а отже, варіації цих коефіцієнтів можуть відповідати змінам в аудіоконтенті, які сприймаються глядачами як значні.

— **3 з 5-бітової гістограми максимального середньоквадратичного відношення для кожного триплету RGB**: Ця функція

також демонструє невелике позитивне зміщення близько 2-3%. Це співвідношення фіксує відносну інтенсивність найбільш домінуючого кольору в кадрі. Вище значення може вказувати на більш динамічні сцени, які можуть сприйматися як важливіші або цікавіші, звідси і позитивний вплив.

— **Спектральний розкид STD**: Ця функція представляє чіткий розподіл, де високі значення тяжіють до 1, а низькі - до 0. Цей позитивний зсув свідчить про те, що кадри з різноманітним частотним вмістом (що свідчить про широкий спектр спектральних компонентів), як правило, є більш значущими в оповіді відео.

— **Можливість нахилу камери**: Ця особливість має негативний вплив приблизно на 5-7%. Це може бути пов'язано з тим, що такі рухи камери, як нахили, часто відбуваються під час переходів або менш важливих сцен, а отже, вони з меншою ймовірністю роблять внесок в основний наратив.

— **Кількість виявлених облич**: Цей показник також має негативну тенденцію. Цілком можливо, що сцени з великою кількістю облич не обов'язково роблять внесок у ключові моменти оповіді, оскільки вони можуть представляти масові сцени або другорядних персонажів.

— **Середня абсолютна різниця між 2 послідовними кадрами у відтінках сірого**: Ця ознака демонструє явне негативне упередження. Швидкі зміни у відтінках сірого часто можуть відображати швидкий розвиток подій або перехідні сцени, які можуть не містити ключової інформації для стиснення.

— **STD MFCC 5**: Ця ознака демонструє найбільш значне позитивне зміщення, ймовірно, з тих самих причин, що й $STD\ MFCC3$. Оскільки KNN використовує метрику відстані для своїх прогнозів, варіації в MFCC можуть призвести до появи окремих звукових кла-

стерів, які модель пов'язує з важливим контентом.

— **1, 4 з 5-бітової гістограми максимального середньоквадратичного відношення для кожного триплету RGB**: Ці характеристики мають позитивне зміщення, що потенційно пов'язано з підходом "найближчого сусіда"моделі KNN, де подібні співвідношення інтенсивності кольору можуть допомогти ідентифікувати подібний вміст.

— **Довжина зйомки кадру**: Ця ознака має високу позитивну похибку. Модель BRF може асоціювати довші кадри з ключовим контентом, оскільки вони дають більше часу для розгортання важливих подій.

— **Mean Chroma 7**: Цей показник має помітне негативне зміщення. Характеристики кольоровості - це представлення аудіоконтенту в термінах

П'ять ознак з найбільшим негативним впливом мають схожі властивості за розподілом та зміщенням для обох моделей. На противагу, ознаки, які мають позитивне зміщення суттєво відрізняються.

ВИСНОВКИ

У нашому дослідженні ми досягли хороших результатів в резюмуванні відео завдяки ретельному налаштуванню параметрів моделей. Найкраще показала себе модель Random Balanced Decision Forest, яка при оптимізації за показником F1-Score досягла точності 0,63215, F1-Score - 0,52214, а AUC-ROC - 0,62593. При оптимізації за точністю (accuracy) цієї моделі показник точності зростає до 0,69689, проте інші метрики спадають до F1-Score - 0.29524, а AUC-ROC - 0.56921. У той же час, модель K-найближчого сусіда, оптимізована для F1-Score, показує точність 0,61052, F1-Score - 0,42867, а AUC-ROC - 0,56783. Коли ця модель була оптимізована для точності (accuracy), вона досягла результату 0,66721, при чому F1-Score - 0.32947, а AUC-ROC - 0.57068.

Згодом ми застосували різні методи для оцінки важливості ознак, що призвело до визначення 121 ознаки для моделі дерева рішень і 130 для підходу KNN. Застосування кореляції Спірмена та ітеративного виключення ознак було інструментом у цій процедурі.

Що до важливості характеристик, найбільш значущими для обох моделей виявилися: середня абсолютна різниця між послідовними кадрами у відтінках сірого, що вказує на динамічний контент; тривалість поточного кадру, що відображає його важливість; ймовірність нахилу камери, що підкреслює певні елементи відео; розподіл інтенсивності червоного кольору; кількість виявлених особин; стандартне відхилення третього цепстрального коефіцієнта Mel-частоти, що фіксує звукові події.

З іншого боку, для обох моделей були визначені ознаки з найменшим впливом: середня надійність виявлення та співвідношення площ для тварин, транспортних засобів і вулиць; кількість виявлених тварин і транспортних засобів; середня надійність виявлення людей. Ці ознаки були менш важливими для узагальнення відео, ймовірно, через їхню низьку кореляцію з ключовими моментами відео. Цікаво, що хоча кількість виявлених осіб була важливою, достовірність виявлення не зробила значного внеску в моделі, що свідчить про те, що сама присутність людей є більш важливою для ідентифікації ключових сегментів відео.

Крім того, ми помітили, що певні ознаки мають вагомий точковий вплив. Для обох моделей ознаки зі значним позитивним впливом включали: певні біни триплетних гістограм RGB, що вказують на зміну інтенсивності кольору; тривалість поточного кадру; кількість виявлених осіб; певні біни гістограм червоного і синього кольорів; певні біни гістограм зеленого-червоного кольорів; середня енергетична ентропія. Ці результати свідчать про те, що нюанси в інтенсивності кольору, тривалості кадру, присутності людей і складності звуку відіграють вирішальну роль у визначенні ключових сегментів відео, причому їхня значущість сильно залежить від контексту.

Ознаки з значним негативним впливом виявилися кількість виявлених людей, тривалість поточного кадру, 1 з 8-бітної гістограми значень насиченості, кількість виявлених облич і середня абсолютна різниця між двома послідовними кадрами в градаціях сірого. Ці особливості проілюстрували, що сцени з великим натовпом, довгі кадри, високо насичені кадри або кадри з великою кількістю облич не обов'язково є важливими для узагальнення. Ключові події або переходи можуть відбуватися в менш людних, коротших або менш

візуально насичених кадрах, що підкреслює необхідність точкової інтерпретації ознак.

Підсумовуючи, наше комплексне дослідження систематичного зміщення виявило певні ознаки, які демонструють стійкі закономірності у своєму впливі на моделі. Такі характеристики, як STD $MFCC_3$, триплетна гістограма RGB та спектральний розкид STD, зазвичай позитивно впливають на важливість кадру в обох моделях. Натомість можливість нахилу камери, кількість виявлених облич та абсолютна різниця між послідовними кадрами у відтінках сірого, як правило, мають негативний вплив. Функція STD моделі $MFCC_5$ продемонструвала найбільш виражене позитивне зміщення, тоді як функція $Mean$ моделі $Chroma_7$ продемонструвала значне негативне зміщення.

РОЗДІЛ 5
ПЛАНИ НА МАЙБУТНЄ ДОСЛІДЖЕННЯ

Прагнучи вдосконалити покращити методологію мультимодального відеоаналізу, ми визначили кілька ключових напрямків для майбутніх досліджень і розробок. Ці напрямки ґрунтуються на наших поточних результатах і покликані вирішити деякі обмеження та питання, і все ще залишаються відкритими. Наше дослідження було зосереджене насамперед на мультимодальному підході до використання даних. Однак, була б цікаво отримати повне уявлення про те, як змінюється важливість ознак, можна провести порівняння результатів отриманих нами та якщо використовувати унімодальні підходи. Проводячи порівняльні експерименти, цілю буде виявити будь-які важливі ознаки, які могли бути пропущені або недооцінені в підході мультимодальної моделі. Це також дасть можливість оцінити можливі потенційні переваги унімодального підходу. Іншим важливим напрямком може бути застосування глибоких нейронних мереж для стиснення відео або інших ансамблевих методів, наприклад XGBoost. В роботі було використано керовану непараметричну модель лісу випадкових збалансованих рішень і некеровану параметричну модель KNN для аналізу мультимодального відео. У майбутніх дослідженнях було б доцільно включити в аналіз глибокі нейронні мережі з їхньою підвищеною складністю та архітектурною різноманітністю, для розуміння чи це підвищити точність і ефективність аналізу мультимодального відео. Одним із цікавих напрямків для майбутнього дослідження є використання трансформерної архітектури для мультимодального відео сумування. Ми плануємо

дослідити, як трансформери можуть бути використані для моделювання взаємодії між різними модальностями у відео та визначення важливості ознак. Це дозволить нам зрозуміти, як трансформери порівняно з іншими архітектурами впливають на точність та роботу мультимодального аналізу відео. Для подальшого покращення мультимодального аналізу відео ми плануємо розширити дослідження визначивши важливості ознак на більш широкому обсязі даних. Ми спробуємо збільшити та покращити використаний набір даних, додавши нові відео, які охоплюють нові тематики, контексти і різноплановість даних. Це дозволить нам отримати більш об'єктивну оцінку важливості ознак і з'ясувати, які фактори можуть впливати на їх значущість у мультимодальному аналізі відео. Враховуючи вищезазначені плани, ми сподіваємося, що наші майбутні дослідження та розробки приведуть до значного покращення в області мультимодального відеоаналізу. Ми віримо, що наші зусилля допоможуть допоможуть вирішити існуючі обмеження та питання, а також відкрити нові можливості для досліджень у цій області.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Ghauri, J. A., Hakimov, S., Ewerth, R. (2021). Supervised video summarization via multiple feature sets with parallel attention. *Proceedings - IEEE International Conference on Multimedia and Expo*. doi:10.1109/ICME51207.2021.9428318
2. Psallidas, T., Koromilas, P., Giannakopoulos, T., Spyrou, E. (2021). Multimodal summarization of user-generated videos. *Applied Sciences (Switzerland)*, *11*(11). doi:10.3390/app11115260
3. Jung, Y., Cho, D., Kim, D., Woo, S., Kweon, I. S. (2019). Discriminative feature learning for unsupervised video summarization. *33rd AAAI Conference on Artificial Intelligence, AAAI 2019, 31st Innovative Applications of Artificial Intelligence Conference, IAAI 2019 and the 9th AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019*, 8537-8544.
4. He, X., Hua, Y., Song, T., Zhang, Z., Xue, Z., Ma, R., ... Guan, H. (2019). Unsupervised video summarization with attentive conditional generative adversarial networks. *MM 2019 - Proceedings of the 27th ACM International Conference on Multimedia*, 2296-2304. doi:10.1145/3343031.3351056
5. Shingrakhia, H., Patel, H. (2020). Emperor penguin optimized event recognition and summarization for cricket highlight generation. *Multimedia Systems*, *26*(6), 745-759. doi:10.1007/s00530-020-00684-3
6. Evangelopoulos, G., Zlatintsi, A., Potamianos, A., Maragos, P., Rapantzikos, K., Skoumas, G., Avrithis, Y. (2013). Multimodal saliency and fusion for movie summarization based on aural, visual, and

textual attention. *IEEE Transactions on Multimedia*, *15*(7), 1553-1568. doi:10.1109/TMM.2013.2267205

7. Lu, S., Wang, Z., Mei, T., Guan, G., Feng, D. D. (2014). A bag-of-importance model with locality-constrained coding based feature learning for video summarization. *IEEE Transactions on Multimedia*, *16*(6), 1497-1509. doi:10.1109/TMM.2014.2319778
8. Precision and recall. URL: ung2019 Jung, Y., Cho, D., Kim, D., Woo, S., Kweon, I. S. (2019). Discriminative feature learning for unsupervised video summarization. *33rd AAAI Conference on Artificial Intelligence, AAAI 2019, 31st Innovative Applications of Artificial Intelligence Conference, IAAI 2019 and the 9th AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019*, 8537-8544.
9. He, X., Hua, Y., Song, T., Zhang, Z., Xue, Z., Ma, R., ... Guan, H. (2019). Unsupervised video summarization with attentive conditional generative adversarial networks. *MM 2019 - Proceedings of the 27th ACM International Conference on Multimedia*, 2296-2304. doi:10.1145/3343031.3351056
10. Shingrakhia, H., Patel, H. (2020). Emperor penguin optimized event recognition and summarization for cricket highlight generation. *Multimedia Systems*, *26*(6), 745-759. doi:10.1007/s00530-020-00684-3
11. Evangelopoulos, G., Zlatintsi, A., Potamianos, A., Maragos, P., Rapantzikos, K., Skoumas, G., Avrithis, Y. (2013). Multimodal saliency and fusion for movie summarization based on aural, visual, and textual attention. *IEEE Transactions on Multimedia*, *15*(7), 1553-1568. doi:10.1109/TMM.2013.2267205
12. Lu, S., Wang, Z., Mei, T., Guan, G., Feng, D. D. (2014). A bag-of-importance model with locality-constrained coding based feature

learning for video summarization. *IEEE Transactions on Multimedia*, *16*(6), 1497-1509. doi:10.1109/TMM.2014.2319778

13. Precision and recall. URL: ung2019 Jung, Y., Cho, D., Kim, D., Woo, S., Kweon, I. S. (2019). Discriminative feature learning for unsupervised video summarization. *33rd AAAI Conference on Artificial Intelligence, AAAI 2019, 31st Innovative Applications of Artificial Intelligence Conference, IAAI 2019 and the 9th AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019*, 8537-8544.
14. He, X., Hua, Y., Song, T., Zhang, Z., Xue, Z., Ma, R., ... Guan, H. (2019). Unsupervised video summarization with attentive conditional generative adversarial networks. *MM 2019 - Proceedings of the 27th ACM International Conference on Multimedia*, 2296-2304. doi:10.1145/3343031.3351056
15. Shingrakhia, H., Patel, H. (2020). Emperor penguin optimized event recognition and summarization for cricket highlight generation. *Multimedia Systems*, *26*(6), 745-759. doi:10.1007/s00530-020-00684-3
16. Evangelopoulos, G., Zlatintsi, A., Potamianos, A., Maragos, P., Rapantzikos, K., Skoumas, G., Avrithis, Y. (2013). Multimodal saliency and fusion for movie summarization based on aural, visual, and textual attention. *IEEE Transactions on Multimedia*, *15*(7), 1553-1568. doi:10.1109/TMM.2013.2267205
17. Lu, S., Wang, Z., Mei, T., Guan, G., Feng, D. D. (2014). A bag-of-importance model with locality-constrained coding based feature learning for video summarization. *IEEE Transactions on Multimedia*, *16*(6), 1497-1509. doi:10.1109/TMM.2014.2319778
18. Precision and recall. URL: https://en.wikipedia.org/wiki/Precision_and_recall Precision and recall. URL: https://en.

wikipedia.org/wiki/Precision_and_recall

19. Hucker Marius - Tree algorithms explained: Ball Tree Algorithm vs. KD Tree vs. Brute Force. URL: https://towardsdatascience.com/tree-algorithms-explained-ball-tree-algorithm-vs-kd-tree-vs-b
20. Milev A. KD Tree - Searching in N-dimensions, Part I. CodeProject - For those who code. URL: https://www.codeproject.com/Articles/18113/KD-Tree-Searching-in-N-dimensions-Part-I.
21. 1.6. Nearest Neighbors. scikit-learn. URL: https://scikit-learn.org/stable/modules/neighbors.html.
22. 1.6. Nearest Neighbors. scikit-learn. URL: https://scikit-learn.org/stable/modules/neighbors.html.
23. 1.10. Decision Trees. scikit-learn. URL: https://scikit-learn.org/stable/modules/tree.html.

ДОДАТОК 1. СПИСОК ЗІ 121 ОЗНАКИ, ЯКІ ЗАЛИШИЛИСЯ ПІСЛЯ КЛАСТЕРИЗАЦІЇ ДЛЯ RANDOM BALANCED DECISION FOREST НА ОСНОВІ РАНГОВИХ КОРЕЛЯЦІЙ СПІРМЕНА ТА ІТЕРАТИВНОГО ВИЛУЧЕННЯ ОЗНАК НА БАЗІ КЛАСТЕРІВ КОРЕЛЯЦІЇ ТА МЕТОДУ ВИЛУЧЕННЯ СТОВПЦІВ:

1. Mean Energy
2. Mean Entropy of Energy
3. Mean Spectral Spread
4. Mean Spectral Flux
5. Mean $MFCC_2$
6. Mean $MFCC_3$
7. Mean $MFCC_4$
8. Mean $MFCC_5$
9. Mean $MFCC_6$
10. Mean $MFCC_8$
11. Mean $MFCC_9$
12. Mean $MFCC_{12}$
13. Mean $MFCC_{13}$
14. Mean $Chroma_1$
15. Mean $Chroma_2$
16. Mean $Chroma_3$
17. Mean $Chroma_4$
18. Mean $Chroma_5$
19. Mean $Chroma_6$

20. Mean $Chroma_7$
21. Mean $Chroma_8$
22. Mean $Chroma_9$
23. Mean $Chroma_{10}$
24. Mean $Chroma_{11}$
25. Mean $Chroma_{12}$
26. Mean Chroma Deviation
27. Mean Zero Crossing Rate Delta
28. Mean Energy Delta
29. Mean Entropy of Energy Delta
30. Mean Spectral Centroid Delta
31. Mean Spectral Spread Delta
32. Mean Spectral Entropy Delta
33. Mean Spectral Flux Delta
34. Mean $MFCC_1$ Delta
35. Mean $MFCC_2$ Delta
36. Mean $MFCC_3$ Delta
37. Mean $MFCC_4$ Delta
38. Mean $MFCC_5$ Delta
39. Mean $MFCC_6$ Delta
40. Mean $MFCC_7$ Delta
41. Mean $MFCC_8$ Delta
42. Mean $MFCC_9$ Delta
43. Mean $MFCC_{10}$ Delta
44. Mean $MFCC_{11}$ Delta
45. Mean $MFCC_{12}$ Delta
46. Mean $MFCC_{13}$ Delta
47. Mean $Chroma_1$ Delta
48. Mean $Chroma_2$ Delta

49. Mean $Chroma_3$ Delta
50. Mean $Chroma_4$ Delta
51. Mean $Chroma_5$ Delta
52. Mean $Chroma_6$ Delta
53. Mean $Chroma_7$ Delta
54. Mean $Chroma_8$ Delta
55. Mean $Chroma_9$ Delta
56. Mean $Chroma_{10}$ Delta
57. Mean $Chroma_{11}$ Delta
58. Mean $Chroma_{12}$ Delta
59. Mean Chroma Deviation Delta
60. STD Zero Crossing Rate
61. STD Energy
62. STD Spectral Spread
63. STD $MFCC_1$
64. STD $MFCC_2$
65. STD $MFCC_3$
66. STD $MFCC_4$
67. STD $MFCC_5$
68. STD $MFCC_7$
69. STD $MFCC_8$
70. STD $MFCC_9$
71. STD $MFCC_{10}$
72. STD $MFCC_{11}$
73. STD $MFCC_{13}$
74. STD Chroma Deviation
75. 1 of 8-bin histogram of the red values
76. 2 of 8-bin histogram of the red values
77. 3 of 8-bin histogram of the red values

78. 4 of 8-bin histogram of the red values
79. 5 of 8-bin histogram of the red values
80. 6 of 8-bin histogram of the red values
81. 7 of 8-bin histogram of the red values
82. 8 of 8-bin histogram of the red values
83. 1 of 8-bin histogram of the green values
84. 2 of 8-bin histogram of the green values
85. 3 of 8-bin histogram of the green values
86. 4 of 8-bin histogram of the green values
87. 5 of 8-bin histogram of the green values
88. 6 of 8-bin histogram of the green values
89. 7 of 8-bin histogram of the green values
90. 3 of 8-bin histogram of the blue values
91. 4 of 8-bin histogram of the blue values
92. 5 of 8-bin histogram of the blue values
93. 6 of 8-bin histogram of the blue values
94. 7 of 8-bin histogram of the blue values
95. 8 of 8-bin histogram of the blue values
96. 6 of 8-bin histogram of the grayscale values
97. 7 of 8-bin histogram of the grayscale values
98. 1 of 5-bin histogram of the max-by-mean-ratio for each RGB triplet
99. 3 of 5-bin histogram of the max-by-mean-ratio for each RGB triplet
100. 4 of 5-bin histogram of the max-by-mean-ratio for each RGB triplet
101. 1 of 8-bin histogram of the saturation values
102. 3 of 8-bin histogram of the saturation values
103. 5 of 8-bin histogram of the saturation values
104. 6 of 8-bin histogram of the saturation values
105. Average absolute difference between two successive frames in grayscale
106. Number of faces detected

107. Average magnitude of the flow vectors
108. Possibility of a camera tilt movement
109. Current shot duration
110. Number of persons detected
111. Persons average detection confidence
112. Average ratio of persons area to the area of the frame
113. Number of vehicles detected
114. Vehicles average detection confidence
115. Average ratio of vehicles area to the area of the frame
116. Number of outdoors detected
117. Outdoors average detection confidence
118. Average ratio of outdoors area to the area of the frame
119. Number of animals detected
120. Animals average detection confidence
121. Average ratio of animals area to the area of the frame

ДОДАТОК 2. СПИСОК ЗІ 130 ОЗНАКИ, ЯКІ ЗАЛИШИЛИСЯ ПІСЛЯ КЛАСТЕРИЗАЦІЇ ДЛЯ МЕТОДУ К-НАЙБЛИЖЧИХ СУСІДІВ (KNN) НА ОСНОВІ РАНГОВИХ КОРЕЛЯЦІЙ

1. Mean Zero Crossing Rate
2. Mean Energy
3. Mean Entropy of Energy
4. Mean Spectral Spread
5. Mean Spectral Flux
6. Mean $MFCC_2$
7. Mean $MFCC_3$
8. Mean $MFCC_4$
9. Mean $MFCC_5$
10. Mean $MFCC_6$
11. Mean $MFCC_7$
12. Mean $MFCC_8$
13. Mean $MFCC_9$
14. Mean $MFCC_{10}$
15. Mean $MFCC_{11}$
16. Mean $MFCC_{12}$
17. Mean $MFCC_{13}$
18. Mean $Chroma_1$
19. Mean $Chroma_2$
20. Mean $Chroma_3$
21. Mean $Chroma_4$
22. Mean $Chroma_5$

23. Mean $Chroma_6$
24. Mean $Chroma_7$
25. Mean $Chroma_8$
26. Mean $Chroma_9$
27. Mean $Chroma_{10}$
28. Mean $Chroma_{11}$
29. Mean $Chroma_{12}$
30. Mean Chroma Deviation
31. Mean Zero Crossing Rate Delta
32. Mean Energy Delta
33. Mean Entropy of Energy Delta
34. Mean Spectral Centroid Delta
35. Mean Spectral Spread Delta
36. Mean Spectral Entropy Delta
37. Mean Spectral Flux Delta
38. Mean $MFCC_1$ Delta
39. Mean $MFCC_2$ Delta
40. Mean $MFCC_3$ Delta
41. Mean $MFCC_4$ Delta
42. Mean $MFCC_5$ Delta
43. Mean $MFCC_6$ Delta
44. Mean $MFCC_7$ Delta
45. Mean $MFCC_8$ Delta
46. Mean $MFCC_9$ Delta
47. Mean $MFCC_{10}$ Delta
48. Mean $MFCC_{11}$ Delta
49. Mean $MFCC_{12}$ Delta
50. Mean $MFCC_{13}$ Delta
51. Mean $Chroma_1$ Delta

52. Mean $Chroma_2$ Delta
53. Mean $Chroma_3$ Delta
54. Mean $Chroma_4$ Delta
55. Mean $Chroma_5$ Delta
56. Mean $Chroma_6$ Delta
57. Mean $Chroma_7$ Delta
58. Mean $Chroma_8$ Delta
59. Mean $Chroma_9$ Delta
60. Mean $Chroma_{10}$ Delta
61. Mean $Chroma_{11}$ Delta
62. Mean $Chroma_{12}$ Delta
63. Mean Chroma Deviation Delta
64. STD Zero Crossing Rate
65. STD Energy
66. STD Spectral Centroid
67. STD Spectral Spread
68. STD $MFCC_1$
69. STD $MFCC_2$
70. STD $MFCC_3$
71. STD $MFCC_4$
72. STD $MFCC_5$
73. STD $MFCC_6$
74. STD $MFCC_7$
75. STD $MFCC_8$
76. STD $MFCC_9$
77. STD $MFCC_{10}$
78. STD $MFCC_{11}$
79. STD $MFCC_{12}$
80. STD $MFCC_{13}$

81. STD Chroma Deviation
82. 1 of 8-bin histogram of the red values
83. 2 of 8-bin histogram of the red values
84. 3 of 8-bin histogram of the red values
85. 4 of 8-bin histogram of the red values
86. 5 of 8-bin histogram of the red values
87. 6 of 8-bin histogram of the red values
88. 7 of 8-bin histogram of the red values
89. 8 of 8-bin histogram of the red values
90. 1 of 8-bin histogram of the green values
91. 2 of 8-bin histogram of the green values
92. 3 of 8-bin histogram of the green values
93. 4 of 8-bin histogram of the green values
94. 5 of 8-bin histogram of the green values
95. 6 of 8-bin histogram of the green values
96. 7 of 8-bin histogram of the green values
97. 3 of 8-bin histogram of the blue values
98. 4 of 8-bin histogram of the blue values
99. 5 of 8-bin histogram of the blue values
100. 6 of 8-bin histogram of the blue values
101. 7 of 8-bin histogram of the blue values
102. 8 of 8-bin histogram of the blue values
103. 6 of 8-bin histogram of the grayscale values
104. 7 of 8-bin histogram of the grayscale values
105. 1 of 5-bin histogram of the max-by-mean-ratio for each RGB triplet
106. 3 of 5-bin histogram of the max-by-mean-ratio for each RGB triplet
107. 4 of 5-bin histogram of the max-by-mean-ratio for each RGB triplet
108. 1 of 8-bin histogram of the saturation values
109. 2 of 8-bin histogram of the saturation values

110. 3 of 8-bin histogram of the saturation values
111. 4 of 8-bin histogram of the saturation values
112. 5 of 8-bin histogram of the saturation values
113. 6 of 8-bin histogram of the saturation values
114. Average absolute difference between two successive frames in grayscale
115. Number of faces detected
116. Average magnitude of the flow vectors
117. Possibility of a camera tilt movement
118. Current shot duration
119. Number of persons detected
120. Persons average detection confidence
121. Average ratio of persons area to the area of the frame
122. Number of vehicles detected
123. Vehicles average detection confidence
124. Average ratio of vehicles area to the area of the frame
125. Number of outdoors detected
126. Outdoors average detection confidence
127. Average ratio of outdoors area to the area of the frame
128. Number of animals detected
129. Animals average detection confidence
130. Average ratio of animals area to the area of the frame

Рецензія
на кваліфікаційну роботу бакалавра на тему:
«Аналіз характеристик мультимодальних керованих моделей для стиснення відео»
студента 4-го курсу факультету комп'ютерних наук та кібернетики
Київського національного університету імені Тараса Шевченка
Чомко Василя Дмитровича

Об'єм інформації, представленої у вигляді відео, зростає набагато швидше за більшість інших видів інформації. У результаті виникає необхідність обробки величезних об'ємів відеоданих. Студент Василь Чомко у своїй роботі вивчає цю проблему, аналізуючи оптимальні візуальні та аудіо характеристики для процесу стиснення відео, яке базується на обробці мультимодальних ознак. Завдяки застосуванню двох методів машинного навчання - Balanced Random Decision Forest та KNN (метод k-найближчих сусідів), він визначає найважливіші характеристики, які необхідні для ефективного стиснення відео.

Перша частина роботи присвячена теоретичному обґрунтуванню алгоритмів, що використовуються в дослідженні. У другій частині дослідження продемонстровано практичне застосування цих алгоритмів, оброблено реальні відеозаписи і дано відповідь на ключове питання роботи: "Чому саме ці характеристики є важливими для стиснення відео?"

Вважаю, що кваліфікаційна робота Василя Чомка відповідає всім вимогам до бакалаврських робіт та заслуговує оцінки "відмінно", а її автор заслуговує на присвоєння кваліфікації бакалавра.

Рецензент:
доктор фізико-математичних наук,
професор Дмитро Номіровський

Відгук
на кваліфікаційну роботу бакалавра на тему:
«Аналіз характеристик мультимодальних керованих моделей для стиснення відео»
студента 4-го курсу факультету комп'ютерних наук та кібернетики Київського національного університету імені Тараса Шевченка
Чомко Василя Дмитровича

Кваліфікаційна робота присвячена покращенню процесу стиснення відео шляхом використання оптимальної комбінації візуальних та аудіо характеристик, враховуючи мультимодальні ознаки. Це покликано забезпечити високу якість відео, що були зняті за допомогою екшн-камер без професійного обладнання чи додаткової постобробки.

Мультимодальне стиснення відео – це область, що потребує глибокого та всебічного дослідження. Визначення набору ключових характеристик, що важливі для стиснення відео, є викликом через велику кількість даних, що мають бути оброблені. У своїй роботі студент Чомко Василь Дмитрович дослідив цю проблему, використовуючи два різних алгоритми машинного навчання: Balanced Random Decision Forest та метод k-найближчих сусідів (KNN). Ці методи були застосовані для виявлення найважливіших спільних характеристик, що дозволяють зменшити обсяг даних для обробки, зберігаючи при цьому точність моделей.

Студент продемонстрував впевнені навички в аналізі та застосуванні теоретичних концепцій на практиці, здатність до самостійного мислення та дослідження. Йому вдалося визначити список важливих об'єднаних характеристик, що допомагає скоротити обсяг даних для обробки, при цьому зберігаючи високий рівень точності моделей.

Робота показує здатність студента аналізувати і застосовувати теоретичний матеріал у практичних завданнях, а також здатність до самостійного мислення та наукового дослідження. Вважаю, що робота відповідає всім критеріям бакалаврських робіт та заслуговує на оцінку "відмінно", а студент Василь Чомко присвоєння кваліфікації бакалавра.

Асистент кафедри обчислювальної математики  Сергій ДЕНИСОВ

UNIVERSITY OF WATERLOO

FACULTY OF ENGINEERING | Systems Design Engineering
519-888-4567, ext. 32600
sydedept@uwaterloo.ca | systems.uwaterloo.ca

June 16, 2023

REVIEW

on the qualification work of the Bachelor of Computer Science

Vasily Dmitrievich Chomko

«Analysis of Characteristics of Multimodal Supervised Models for Video Compression»

It is with pleasure that I report on my assessment of the thesis work of Computational Mathematics student Vasyl Chomko, with whom I have regularly met during his academic stay at the University of Waterloo. I currently have many graduate students working on machine-learning models, with one Master's student specifically working on video chaptering as part of an industry research collaboration, so I believe I have the background necessary to assess the work.

The thesis focused on the study of video chaptering and summarization, specifically on the assessment of multimodal features (visual and/or acoustic), and to try to better understand which features play the most/least significant role in gaining an effective understanding of the video. The student conducted an in-depth analysis of the literature, leading to two substantial pieces of work:

1. An implementation of classifiers (based on Random Forests and CNNs);
2. Feature selection and quantitative assessment of feature relevance / performance.

The latter (second) task was the conceptual focus of the thesis, however it was the former classification task, an essential component of testing the features, which required a great deal of work on the part of the student. Fortunately, both classifiers were implemented successfully and therefore allowed the thesis to progress to the more meaningful question of feature assessment. The results clearly show that the degree of statistical relevance / importance of the various feature vary quite widely, giving concrete information about each clustered feature group.

The thesis identifies limitations and directions for future work, in particular setting the stage for further work whereby much smaller / lightweight methods could be proposed for the video summarization task, by processing a far smaller set of inputs, focusing only on a reduced set of highly-relevant features, rather than all features or entire video frames.

The assigned tasks were completed on time, meetings with the student were productive, research conversations with the student were engaging and in-depth, and the thesis leads to meaningful conclusions which leave open directions of interest for further work.

In my opinion, the presented thesis meets the requirements for a bachelor's degree in Computer Science. I believe that the work deserves a grade of "Very Good", and Vasily Dmitrievich Chomko has most definitely earned the educational qualification of "Bachelor in Computer Science".

I would be more than happy to be contacted if there were any questions. Sincerely,

Paul Fieguth
Professor, Systems Design Engineering
University of Waterloo

UNICHECK
by Turnitin

Ім'я користувача:
Оноцький В'ячеслав ФКомпНаук

ID перевірки:
1015632634

Дата перевірки:
17.06.2023 13:49:42 EEST

Тип перевірки:
Doc vs Internet + Library

Дата звіту:
17.06.2023 13:55:55 EEST

ID користувача:
100002816

Назва документа: ЧомкоВасильДмитрович

Кількість сторінок: 106 Кількість слів: 19519 Кількість символів: 141984 Розмір файлу: 10.45 MB ID файлу: 1015279206

Виявлено модифікації тексту (можуть впливати на відсоток схожості)

1.8%
Схожість

Найбільша схожість: 0.15% з Інтернет-джерелом (https://ses.library.usyd.edu.au/bitstream/handle/2123/13550/guan_gg_.

1.09% Джерела з Інтернету | 331 ..... Сторінка 108

1.31% Джерела з Бібліотеки | 166 ..... Сторінка 109

0% Цитат

Вилучення цитат вимкнене

Вилучення списку бібліографічних посилань вимкнене

0%
Вилучень

Немає вилучених джерел

Модифікації

Виявлено модифікації тексту. Детальна інформація доступна в онлайн-звіті.

Замінені символи | 6328

Підозріле форматування | 45 сторінок

Експертна оцінка роботи науковим керівником :

Робота студента 4-го курсу, Чомка Василя - ”Аналіз характеристик мультимодальних керованих моделей для стиснення відео” виконана самостійно, при цьому обсяг цитувань та запозичень становить 1.8% та не перевищує норму.

Науковий керівник: _____ (підпис) Денисов С.В. (ПІБ)

Оператор: _____ (підпис) Оноцький В.В. (ПІБ)