УДК 004.8 DOI 10.36910/775.24153966.2021.72.8

А. В. Кошель
Магістр Київського національного університету імені Тараса Шевченка
Розробник програмного забезпечення
Товариство з обмеженою відповідальністю "Арткай"
02000, м. Київ, вул. Генерала Геннадія Воробйова, 13
https://orcid.org/0000-0003-1099-0995

ВПРОВАДЖЕННЯ МАШИННОГО НАВЧАННЯ НА МОБІЛЬНІ ПРИСТРОЇ

У статті розкрито архітектурні складові впровадження машинного навчання на мобільні пристрої. Виділено основні переваги використання глибокого навчання і логічного висновку на мобільному пристрої: економія смуги пропускання зв'язку, зниження вартості ресурсів хмарних обчислень, швидкий час відгуку, мобільні обчислення зберігають сенсорні дані на локальному пристрої, що значно покращує конфіденційність даних користувача. Підкреслено, що на сьогодні існує п'ять архітектур, які зазвичай використовуються для реалізації машинного навчання на мобільних пристроях. При цьому, наголошено, що кожна окрема архітектура, її фундаментальна основа залежить від деталей сценарію, таких як специфічні вимоги мобільного додатку, складність моделі, кількість даних тощо. Перша архітектура – це вивід даних без навчання, ґрунтується на тому, що мобільний додаток надсилає запит до хмари через інтерфейс прикладного програмування разом з новими даними, а служба повертає прогноз. Друга архітектура – це вивід даних та навчання у хмарі, в її основі лежить принцип попередньої моделі, єдина відмінність полягає в тому, що постачальники послуг надають розробникам мобільних пристроїв можливість навчати дані та створювати власні унікальні моделі за допомогою хмарного сервісу. Третя архітектура – це виведення на пристрої з попередньо навченими моделями, принцип реалізації засновано на тому, що попередньо навчена модель завантажується у мобільний додаток, щоб зробити виведення, мобільний додаток запускає всі обчислення виведення локально на пристрої. Четверта архітектура – цевиведення і навчання на пристрої, програма може постійно вчитися на даних та поведінці користувача, а отже, постійно оновлювати моделі та покращувати продуктивність для даного користувача. П'ята архітектура – цегібридна архітектура, в основі лежить принцип коли основна модель навчається в хмарі з використанням великого стандартного набору даних або великої сукупності всіх даних, якими користуються користувачі. Наголошено, що на сьогоднішній день, найпростішим способом включення машинного навчання у мобільний додаток є використання хмарного сервісу, який охоплює функціональні можливості обох складових.

Ключові слова: *платформа, архітектура, мобільні пристрої, інтеграція, хмара, блок навчання, блок обчислення, машинне навчання.*

А. В. Кошель

ВНЕДРЕНИЕ МАШИННОГО ОБУЧЕНИЯ НА МОБИЛЬНЫЕ УСТРОЙСТВА

В статье раскрыты архитектурные составляющие внедрения машинного обучения на мобильные устройства. Выделены основные преимущества использования глубокого обучения и логического вывода на мобильном устройстве: экономия полосы пропускания связи, снижение стоимости ресурсов облачных вычислений, быстрое время отклика, мобильные вычисления хранят сенсорные данные на локальном устройстве, что значительно улучшает конфиденциальность данных пользователя. Подчеркнуто, что на сегодня существует пять архитектур, которые обычно используются для реализации машинного обучения на мобильных устройствах. При этом, отмечается, что каждая отдельная архитектура, ее фундаментальная основа зависит от деталей сценария, таких как специфические требования мобильного приложения, сложность модели, количество данных и тому подобное. Первая архитектура-это вывод данных без обучения, основанный на том, что мобильное приложение отправляет запрос в облако через интерфейс прикладного программирования вместе с новыми данными, а служба возвращает прогноз. Вторая архитектура – это вывод данных и обучения в облаке, в ее основе лежит принцип предыдущей модели, единственное отличие заключается в том, что поставщики услуг предоставляют разработчикам мобильных устройств возможность обучать данные и создавать собственные уникальные модели с помощью облачного сервиса. Третья архитектура – это выведение на устройстве с предварительно обученными моделями, принцип реализации основан на том, что предварительно обученная модель загружается в мобильное приложение, чтобы сделать вывод, мобильное приложение запускает все вычисления вывода локально на устройстве. Четвертая архитектура-это введение и обучение на устройстве, программа может постоянно учиться на данных и поведении пользователя, а следовательно, постоянно обновлять модели и улучшать производительность для данного пользователя. Пятая архитектура – это гибридная архитектура, в основе лежит принцип когда основная модель учится в облаке с использованием большого набора данных или большой совокупности всех данных, которыми пользуются пользователи. Отмечается, что на сегодняшний день, самым простым способом включения машинного обучения в мобильное приложение является использование облачного сервиса, который охватывает функциональные возможности обеих составляющих.

Ключевые слова: *платформа, архитектура, мобильные устройства, интеграция, облако, блок обучения, блок вычисления, машинное обучение.*

A. V. Koshel

MACHINE LEARNING ON MOBILE DEVICESIMPLEMENTING

The article reveals the architectural components of the implementation of machine learning on mobile devices. The main advantages of using deep learning and inference on a mobile device are highlighted: bandwidth savings, reduced cost of cloud computing resources, fast response time, mobile computing stores sensory data on the local device, which significantly improves the confidentiality of user data. It is emphasized that today there are five architectures that are commonly used to implement machine learning on mobile devices. At the same time, it is emphasized that each individual architecture, its fundamental basis depends on the details of the scenario, such as the specific requirements of the mobile application, the complexity of the model, the amount of data and so on. The first architecture is non-learning data output, based on the fact that the mobile application sends a request to the cloud through the application programming interface along with the new data, and the service returns the forecast. The second architecture is data output and cloud learning, based on the principle of the previous model, the only difference being that service providers give mobile device developers the ability to learn data and create their own unique models using the cloud service. The third architecture is output to devices with pre-trained models, the principle of implementation is based on the fact that the pre-trained model is loaded into the mobile application to make output, the mobile application runs all output calculations locally on the device. The fourth architecture is output and learning on the device, the program can constantly learn from the data and user behavior, and therefore, constantly update models and improve performance for this user. The fifth architecture is a hybrid architecture, based on the principle that the basic model is trained in the cloud using a large standard set of data or a large set of all data used by users. It is emphasized that today, the easiest way to include machine learning in a mobile application is to use a cloud service that covers the functionality of both components.

Key words: *platform, architecture, mobile devices, integration, cloud, learning unit, computing unit, machine learning.*

Вступ та постановка проблеми. Враховуючи прориви в технологіях глибокого навчання і штучного інтелекту, варто наголосити на встановленні можливості використання безлічі мобільних додатків з підвищеним рівнем функціоналу. Порівнюючи традиційні парадигми обчислень засновані на мобільних датчиках і хмарні обчислення та глибоке навчання, яке реалізовано на мобільних пристроях, візуалізується низка переваг використання останніх. До цих переваг відносяться низька пропускна здатність зв'язку, невелика вартість ресурсів хмарних обчислень, швидкий час відгуку і поліпшена конфіденційність даних.

Глибоке навчання є ключовим фактором багатьох останніх досягнень в області додатків штучного інтелекту. Технології штучного інтелекту з часом стають повсюдними в мобільних додатках, таких як автоматичне водіння, доступні роботи для дому та більш інтелектуальна особиста допомога на мобільному телефоні. У порівнянні з традиційною парадигмою мобільного зондування і хмарних обчислень, переваги глибокого навчання і логічного висновку на мобільному пристрої полягають в чотирьох аспектах:

1) Економія смуги пропускання зв'язку. Чим більше обчислень виконується на мобільному пристрої, тим менше даних потрібно відправляти у хмару.

2) Зниження вартості ресурсів хмарних обчислень. Вартість обслуговування або навіть оренди ресурсів хмарних обчислень може бути непомірно високою для деяких додатків. Обчислення на мобільних пристроях стають можливими в міру того, як мобільні пристрої стають більш потужними в обчислювальному відношенні.

3) Швидкий час відгуку. Якщо всі обчислення можуть бути виконані локально, не буде накладних витрат на час зв'язку або будь-яких проблем з надійністю сервера. Для деяких додатків, наприклад, в сфері охорони здоров'я і у військовій сфері, цей час відгуку є дуже важливим показником.

4) Мобільні обчислення зберігають сенсорні дані на локальному пристрої, що значно покращує конфіденційність даних користувача. Це особливо актуально для додатків домашньої робототехніки.

Таким чином, дослідження в області глибокого навчання для мобільних і вбудованих пристроїв стали актуальною темою. Підтеми охоплюють аспекти архітектури обладнання, алгоритмічну оптимізацію і вибір мобільних платформ глибокого навчання. По мірі розвитку досліджень і розробок багато інструментів, дані і моделі стають загальнодоступними. Для початківців в області мобільного глибокого навчання обсяг інформації, доступної у зазначеній сфері, може бути величезним.

Аналіз останніх досліджень і публікацій. В останні роки з'являється все більше робіт, в яких описуються механізми та принципи застосування машинного навчання та штучного інтелекту на мобільні пристрої.

Так С. С. Гороховський та О. О. Франків [1] створили фреймворк MLARKit, що дає змогу легко користуватися складними для використання у мобільних пристроях алгоритмами машинного навчання та доповненої реальності, враховуючи їхні особливості. Фреймворк створено максимально гнучким, тож сторонні розробники зможуть максимально задовольнити свої потреби без самостійної реалізації.

Розробка програмного забезпечення доповненої реальності для розпізнавання рухів з використанням технологій SWIFT, ARKIT, COREML запропонували В.В. Туйчев та Д.І. Катєльніков [2]. Авторами підкреслено, що актуальною залишається задача розробки мобільного додатку, який комбінує доповнену реальність та можливості розпізнавання рухів, за допомогою яких здійснюється вплив на віртуальні об'єкти. Саме вирішенню цієї задачі і присвячений розроблений науковцями додаток «Hands Gesture AR».

Д.Г. Косяков та О.В. Ларченко[3] дослідили тенденції розвитку сучасних інформаційних технологій, навели визначення основних понять та зазначили принципи функціонування Інтернету речей.

Актуальну науково-прикладну проблему у галузі інструментального забезпечення біоінформатики – розвиток теоретичних засад, удосконалення методологічної, алгоритмічної та програмно-технічної бази комп'ютерних систем опрацювання біосигналів і даних на основі широкого використання штучних нейронних мереж і технологій глибокого навчання розкрив Ю. В. Хома [4].

Впровадження машинного зору на мобільні пристрої для моніторингу завантаженості автодоріг дослідили низка вчених М. М. Гулковський, Ю.О. Борзов та О.В. Придатко [5]. Авторами проведено огляд технологій та методів для розпізнавання транспортних засобів та їх відслідковування на відео. Розглянуто один із можливих шляхів вирішення даної проблеми. за допомогою засобів машинного зору.

Із зарубіжних авторів варто відзначити такі роботи як: M. Hollemans [6], Bayerl, Sebastian & Frassetto, Tommaso & Jauernig, Patrick & Riedhammer, Korbinian & Sadeghi, Ahmad-Reza & Schneider, Thomas & Stapf, Emmanuel & Weinert, Christian [7], Newnham J. [8], He, Yihui & Lin, Ji & Liu, Zhijian & Wang, Hanrui & Li, Li-Jia & Han, Song [9], Tan, Mingxing & Chen, Bo & Pang, Ruoming & Vasudevan, Vijay & Sandler, Mark & Howard, Andrew & Le, Quoc[10], Wolfensparger D. [11], Zhang, Xiangyu & Zhou, Xinyu & Lin, Mengxiao & Sun, Jian[12], Dai, Xiangfeng & Spasic, Irena & Meyer, B. & Chapman, Samuel & Andres, Frederic[13], Ramu, Arulmurugan & K.R, Sabarmathi & Haldorai, Anandakumar[14], Sandler, Mark & Howard, Andrew & Zhu, Menglong & Zhmoginov, Andrey & Chen, Liang-Chieh [15] та інші.

Проте, враховуючи описані наукові набутки, за темою, питання розкриття принципів впровадження машинного навчання на мобільні пристрої залишається відкритим та потребує детального опрацювання.

Постановка завдання. Розкрити принципи впровадження машинного навчання на мобільні пристрої.

Викладення основного матеріалу дослідження. Впровадження машинного навчання на мобільні пристрої реалізується на основі архітектур. На сьогодні, науковцями відокремлено п'ять архітектур, які зазвичай використовуються для реалізації машинного навчання на мобільних пристроях. Кожна окрема архітектура, її фундаментальна основа залежить від деталей сценарію, таких як специфічні вимоги мобільного додатку, складність моделі, кількість даних тощо.

Перша архітектура – це вивід даних без навчання.

Багато мобільних додатків, які базуються на обчисленні у хмарі, наприклад, платформи машинного навчання як послуги або віддалені сервери формують наступну послідовність: мобільний додаток надсилає запит до хмари через інтерфейс прикладного програмування (API) разом з новими даними, а служба повертає прогноз. Немає необхідності, щоб сам додаток (тобто його розробник) виконував навчання. Наприклад, на рисунку 1 показано принцип розпізнавання зображень, який використовує хмарні обчислення без навчальної вибірки. Зображення надсилається до хмарного сервісу, класифікується в хмарі, а результат розпізнавання надсилається назад на мобільний додаток.

Постачальники послуг оновлюють свої моделі. Коли моделі періодично проходять повторне навчання, мобільні додатки автоматично отримують користь від цих удосконалень. Ця архітектура є найпростішим та найшвидшим рішенням для впровадження машинного навчання

для мобільних додатків, але це рішення працює лише для програм, які обробляють загальновживані дані, оскільки за збір навчальних даних та оновлення відповідає лише постачальник послуг, а не розробник програмних моделей. Будь-які випадки, які ґрунтуються на ідіосинкратичних даних, не можуть скористатися цією архітектурою [12].

Друга архітектура – це вивід даних та навчання у хмарі.

Точні моделі машинного навчання вимагають великих наборів навчальних даних, обробка яких, у свою чергу, споживає значну кількість енергії, пам'яті та часу, яких на мобільних пристроях недостатньо. Тому актуальним є як хмарний висновок, так і навчання.

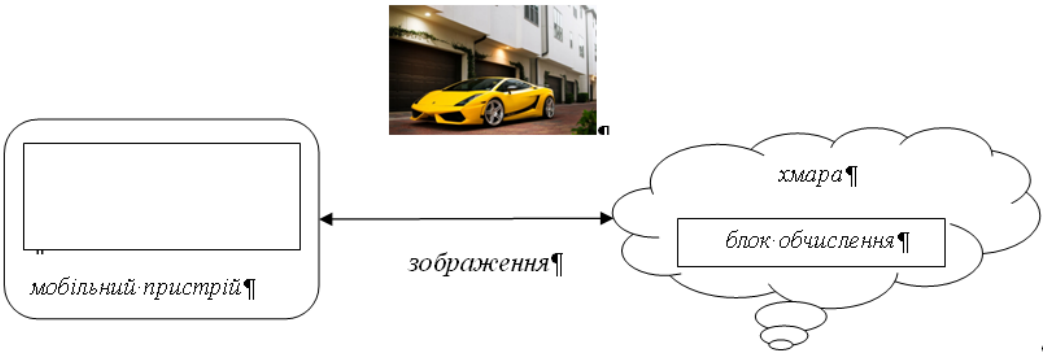

Рис. 1. **Вивід даних без навчання**

Ця архітектура схожа на попередню, єдина відмінність полягає в тому, що постачальники послуг надають розробникам мобільних пристроїв можливість навчати дані та створювати власні унікальні моделі за допомогою хмарного сервісу.

Це забезпечує гнучкість у типах та обсягах даних, які розробник може використовувати для навчання. У наведеному нижче прикладі (рисунок 2) розробник завантажує (не обов'язково з мобільного пристрою) навчальний набір даних (групу зображень, включаючи зображення авто) і використовує їх для навчання моделі в хмарі. Користувач надсилає зображення авто з телефону, а вивід результату виконується в хмарі, а потім надсилається назад на телефон користувача. Надсилання даних користувачів у хмару збільшує проблеми конфіденційності та безпеки, особливо якщо вони зберігаються там для розширення навчального набору даних та повторного навчання моделей, надісланих на сервер.

Третя архітектура – це виведення на пристрої з попередньо навченими моделями.

Виведення на пристрої є важливим для мобільних додатків, де затримка тривалістю кілька мікросекунд є критичною для виконання місії. Час відгуку є основною причиною здійснення виведення результату безпосередньо на пристрої. Наприклад, виведення, яке виконують автомобілі, що їздять самостійно, повинно бути майже миттєвим і незалежним від з'єднання з хмарою. У цій архітектурі попередньо навчена модель завантажується у мобільний додаток. Щоб зробити виведення, мобільний додаток запускає всі обчислення виведення локально на пристрої. Йому не потрібно з'єднання з сервером для будь-чого, що стосується машинного навчання. Виведення відбувається майже миттєво (рисунок 3).

Попередньо навчена модель може бути або стандартною загальною моделлю, такою як [11], [14], або індивідуальною. Якщо стандартна попередньо навчена модель задовольняє вимогам програми (наприклад, розмір моделі), розробникам потрібно лише завантажити попередньо навчену модель у додаток. В іншому випадку потрібна індивідуальна модель. Модель можна попередньо навчити та налаштувати на настільному комп'ютері, високопродуктивному комп'ютерному кластері або в хмарі.

Виведення на пристрої підходить для програм, де проблеми конфіденційності викликають занепокоєння, наприклад, у випадку медичної діагностики. Наприклад, [13] демонструє додаток виведення на пристрої для виявлення раку шкіри. Спочатку модель класифікації попередньо навчається та налаштовується на комп'ютері, потім зберігається на мобільному пристрої. Коли користувач надає новий образ своєї шкіри, він зберігається на його мобільному пристрої, де модель використовується для класифікації ураження шкіри. Результати сповіщення повинні передаватися зовні. Ця архітектура зменшує затримку, економить пропускну здатність та

покращує конфіденційність. Однак масштабні моделі не можуть бути розміщені на мобільному пристрої, розмірність може вплинути на точність та гнучкість використання.

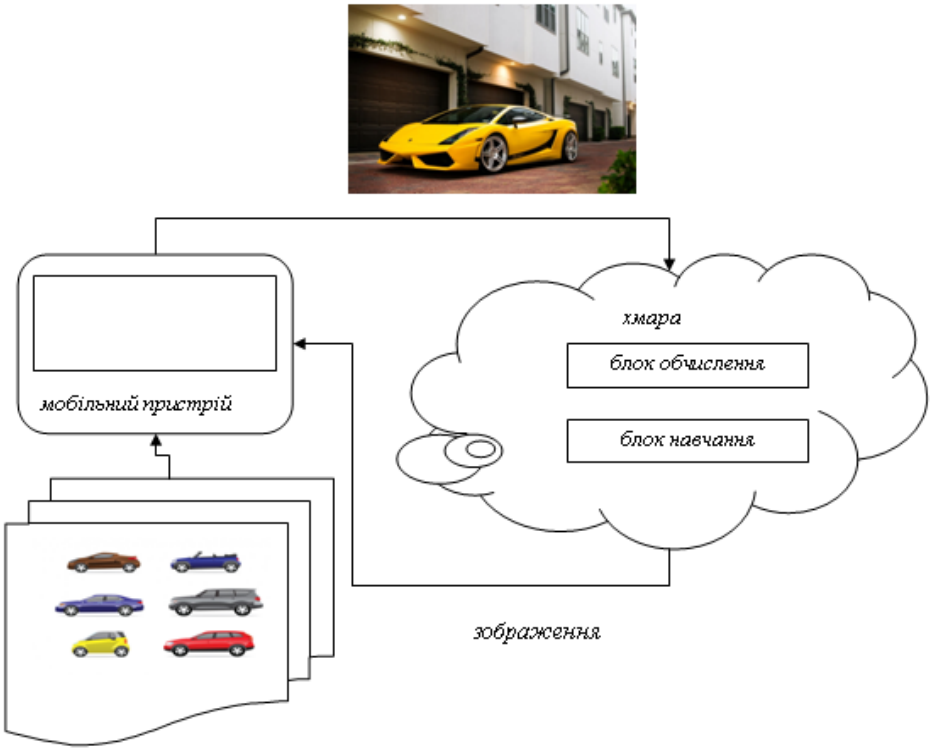

Рис. 2. **Вивід даних та навчання у хмарі**

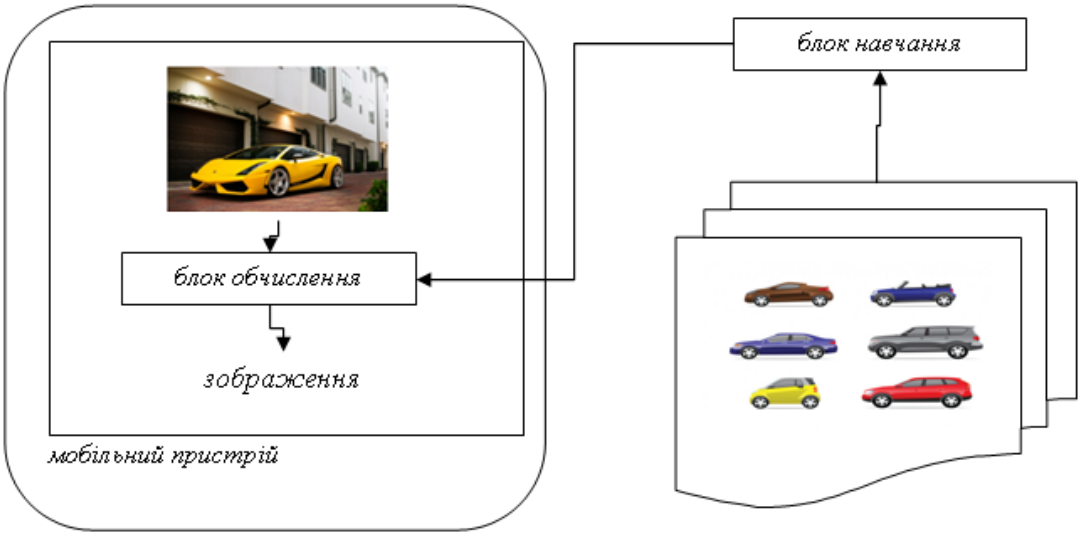

Рис. 3. **Виведення на пристрої з попередньо навченими моделями**

Четверта архітектура – це виведення і навчання на пристрої.

Основною перевагою цієї архітектури є те, що програма може постійно вчитися на даних та поведінці користувача, а отже, постійно оновлювати моделі та покращувати продуктивність для даного користувача.

Ця архітектура економить витрати на використання хмарних сервісів та потребу у пропускній здатності. Насправді, якщо всі дані доступні на пристрої користувача для навчання моделі та виведення інформації, все можна зробити на пристрої, не використовуючи хмари. Ця архітектура можлива для деяких сценаріїв за допомогою ручних глибоких нейронних мереж невеликого розміру [15]. Однак ця архітектура буде працювати лише для невеликих наборів даних

та базових алгоритмів машинного навчання через обмежену доступність пам'яті та обчислень. Інтенсивне навчання на мобільному пристрої залишається недоцільним для більшості застосувань.

П’ята архітектура – це гібридна архітектура.

У цій архітектурі навчання відбувається як на мобільному пристрої, так і в хмарі. Основна модель навчається в хмарі з використанням великого стандартного набору даних або великої сукупності всіх даних, якими користуються користувачі. Окрему модель можна адаптувати для кожного користувача, використовуючи власні дані на власному пристрої для додаткового навчання. Ця гібридна архітектура дозволяє розробникам продовжувати навчання для вдосконалення індивідуальної моделі та адаптації індивідуальної моделі шляхом пристосування моделі до індивідуальних даних користувача. Це покращує індивідуальні вимоги користувачів за допомогою індивідуальних моделей. Ця архітектура є складною, непростою для реалізації та досить дорогою в обслуговуванні.

Більшість популярних хмарних сервісів надають інфраструктуру для попередньої обробки даних, навчання моделі, оцінки моделі та подальшого прогнозування для підтримки хмарних архітектурних додатків. Додатки збирають дані з телефону, надсилають їх у хмару, а потім застосовують машинне навчання до хмари. На сьогоднішній день, найпростішим способом включення машинного навчання у мобільний додаток є використання хмарного сервісу, який охоплює функціональні можливості обох складових.

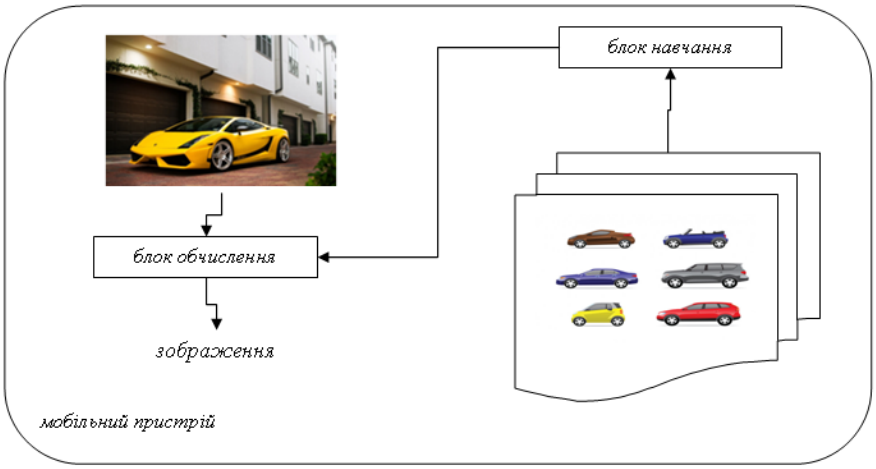

Рис. 4. **Виведення та навчання на пристрої**

Висновки. У роботі розкрито принципи впровадження машинного навчання на мобільні пристрої. Нинішні обмеження обчислювальної потужності, затримок, ризиків конфіденційності та обмеженої пам'яті створюють труднощі та можливості для впровадження машинного навчання на мобільні додатки. З урахуванням обмежень, запропоновано п’ять архітектур для впровадження машинного навчання на мобільних пристроях, окресливши плюси та мінуси кожної з них. Хмарні архітектури є популярним способом впровадження машинного навчання на мобільних пристроях, але технологічна індустрія розвивається з експоненціальною швидкістю, і машинне навчання на пристроях, незабаром, може стати стандартом у мобільних додатках. Наразі повне розгортання виведення інформації та навчання на пристроях все ще нереальне, та й не є необхідним. Зрештою, відповідна архітектура залежить від конкретного сценарію реалізації. Нарешті, зменшення затримок, посилена безпека, можливості роботи в автономному режимі та скорочення витрат визначатимуть розвиток новаторських та інноваційних програм мобільного машинного навчання, які необхідні для подальшого розвитку та вдосконалення.

Література

1. Машинне навчання та доповнена реальність на пристроях на базі iOS із фреймворком MLARKit [Електронний ресурс] / С. С. Гороховський, О. О. Франків // Наукові записки НаУКМА. Комп'ютерні науки. – 2020. – Т. 3. – С. 4-6. – Режим доступу: http://nbuv.gov.ua/UJRN/NaUKMAkn_2020_3_4

2. Туйчев В.В., Катєльніков Д.І. Розробка програмного забезпечення доповненої реальності для розпізнавання рухів з використанням технологій SWIFT, ARKIT, COREML / Тези доповідей II Всеукраїнської науково-технічної конференції «Комп’ютерні технології: інновації, проблеми, рішення», м. Житомир, 14 – 15 листопада 2019 р. – Житомир: Житомирська політехніка, 2019. – С. 39-40.

3. Тенденції розвитку сучасних інформаційних технологій. Косяков Д. Г., Ларченко О.В. – Режим доступу. – URI: http://hdl.handle.net/123456789/6191.
4. Хома Ю.В. Теорія і методи комп'ютерного опрацювання біосигналів на основі машинного навчання. – На правах рукопису. Дисертація на здобуття наукового ступеня доктора технічних наук за спеціальністю 05.13.05 – комп'ютерні системи та компоненти – Національний університет «Львівська політехніка», МОН України, Львів, 2020. 379 с.
5. Система моніторингу завантаженості автодоріг / М. М. Гулковський, Ю.О. Борзов, О.В. Придатко //Матеріали Десятої Міжнародної наукової конференції студентів та молодих вчених «Сучасні інформаційні технології - 2020» «Modern Information Technology - 2020» (14-15 травня 2020 р., м.Одеса) / МОН України; Одес. Нац. політех. ун-т ; Ін-т комп'ют. систем. – Одеса : Наука і техніка, 2020. – С. 120-121.
6. Hollemans M. Core ML Survival Guide [Electronic resource] / Matthiijs Hollemans. – 2018. – 363 p. – Mode of access: https:// leanpub.com/coreml-survival-guide.
7. Bayerl, Sebastian & Frassetto, Tommaso & Jauernig, Patrick & Riedhammer, Korbinian & Sadeghi, Ahmad-Reza & Schneider, Thomas & Stapf, Emmanuel & Weinert, Christian. (2020). Offline Model Guard: Secure and Private ML on Mobile Devices. 460-465. 10.23919/DATE48585.2020.9116560.
8. Newnham J. Machine Learning with Core ML: An iOS developer's guide to implementing machine learning in mobile apps [Electronic resource] / Joshua Newnham. – 2018. – 378 p. – Mode of access: https://www.amazon.com/Machine-Learning-Core-developers-implementing/dp/1788838297.
9. He, Yihui & Lin, Ji & Liu, Zhijian & Wang, Hanrui & Li, Li-Jia & Han, Song. (2018). AMC: AutoML for Model Compression and Acceleration on Mobile Devices. 10.1007/978-3-030-01234-2_48.
10. Tan, Mingxing & Chen, Bo & Pang, Ruoming & Vasudevan, Vijay & Sandler, Mark & Howard, Andrew & Le, Quoc. (2019). MnasNet: Platform-Aware Neural Architecture Search for Mobile. 2815-2823. 10.1109/CVPR.2019.00293.
11. Wolfensparger D. Apple Arkit Revealed: Augmented and Mixed Reality for Iphone and Ipad / Dell Wolfensparger. – 2018. – 180 p.
12. Zhang, Xiangyu & Zhou, Xinyu & Lin, Mengxiao & Sun, Jian. (2018). ShuffleNet: An Extremely Efficient Convolutional Neural Network for Mobile Devices. 6848-6856. 10.1109/CVPR.2018.00716.
13. Dai, Xiangfeng & Spasic, Irena & Meyer, B. & Chapman, Samuel & Andres, Frederic. (2019). Machine Learning on Mobile: An On-device Inference App for Skin Cancer Detection. 10.1109/FMEC.2019.8795362.
14. Ramu, Arulmurugan & K.R, Sabarmathi & Haldorai, Anandakumar. (2019). Classification of sentence level sentiment analysis using cloud machine learning techniques. Cluster Computing. 22. 10.1007/s10586-017-1200-1.
15. Sandler, Mark & Howard, Andrew & Zhu, Menglong & Zhmoginov, Andrey & Chen, Liang-Chieh. (2018). Inverted Residuals and Linear Bottlenecks: Mobile Networks forClassification, Detection and Segmentation.