

МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
**«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИМЕНИ Н. Г. ЧЕРНЫШЕВСКОГО»**

Кафедра математической кибернетики и компьютерных наук

**РАЗРАБОТКА ПРИЛОЖЕНИЯ НА ANDROID ДЛЯ
КЛАССИФИКАЦИИ ЗМЕЙ**

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

студента 4 курса 411 группы
направления 02.03.02 — Фундаментальная информатика и информационные
технологии
факультета КНиИТ
Савина Егора Ивановича

Научный руководитель
к. ф.-м. н., доцент

Ю. Н. Кондратова

Заведующий кафедрой
к. ф.-м. н., доцент

С. В. Миронов

Саратов 2023

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	3
1 Основное содержание работы	4
ЗАКЛЮЧЕНИЕ	10
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	11

ВВЕДЕНИЕ

Машинное обучение и нейронные сети применяются во многих областях человеческой деятельности — лежат в основе машинного перевода, чат-ботах и голосовых помощников, генерируют и обрабатывают фото, видео и музыку.

Обученные нейронные сети можно использовать в мобильных приложениях, что облегчает деятельность людей, помогает повысить эффективность работы приложений и усовершенствовать пользовательский опыт. При этом, использование нейронных сетей в приложениях на смартфонах повышает мобильность самого приложения и дает ему доступность для использования не только с портативного компьютера, а так же и с мобильного устройства.

Целью данной работы является разработка ANDROID приложения для классификации различных видов змей. Для достижения этой цели будет использоваться нейронная сеть, которая будет обучена на наборе данных "Identifying different Breeds of Snakes". Этот набор содержит большое количество изображений змей различных видов.

В связи с поставленной целью необходимо решить следующие задачи:

- Рассмотреть нейронные сети предрасположенных для классификации изображений;
- Изучить инструменты для создания мобильных приложений для ОС ANDROID;
- Обучить нейронную сеть;
- Разработать мобильное приложение на основе обученной нейронной сети.

Работа состоит из введения, 2 глав, заключения, списка из 21 источника, 3 приложений, в которых приведен код программного продукта.

1 Основное содержание работы

Данная дипломная работа начинается с введения, где рассматривается вопрос об актуальности Нейронных сетей, ставится цель дипломной работы и задачи, которые необходимо решить.

Первый раздел «Средства и инструменты для создания приложения» представляет собой реферативный обзор основных работ, в которых отражаются научные представления о свёрточных нейронных сетях, их архитектуре, а также описаны инструменты для работы с нейронными сетями и для создания приложения на ОС Android (трудов таких ученых, как Х. Саймон, И.В. Зайцев, П.Г. Круг и другие).

В подраздел 1.1 «Архитектура свёрточных нейронных сетей» описывается понятие модели свёрточной нейронной сети и приводятся примеры известных свёрточных нейронных сетей.

Свёрточная нейронная сеть — класс нейронных сетей, который специализируется на обработке изображений и видео. Идея, заключённая в свёрточных нейронных сетях, состоит в попеременном использовании субдискретизирующих и свёрточных слоёв.

Свёрточные нейронные сети состоят из нескольких слоёв. Чем больше слоёв, тем мощнее архитектура и лучше обучение нейросети. Основные элементы свёрточной нейронной сети:

- свёрточный слой,
- пулинг,
- полносвязный слой.

На свёрточном слое нейронная сеть удаляет лишнее и оставляет полезное — то, что поможет проанализировать изображение. Например, линии, края или ровные области. Свёртку можно создавать для каждого признака. Нейросеть будет сама подбирать их во время распознавания и классификации на каждом свёрточном слое.

После свёрточного слоя идёт слой пулинга. Из признаков, которые выделил свёрточный слой, выбирает самые важные, а несущественные удаляет. К результату, который получился во время пулинга, можно снова применить свёрточный слой и сделать несколько циклов. Это нужно, чтобы выстроить иерархию признаков: от самых примитивных, например кусочков контура, до более сложных, например глаз человека или форму лица.

Первые слои нейросети анализируют мелкие детали изображения — ворсинки, трещины. Размер изображения может быть 2*2 или 3*3 пикселя — по такому маленькому контексту сложно выделить глаза, форму лица или деревья, на фоне которых стоит человек, но можно найти цвета или перепады света, то есть границы разных объектов. На следующих слоях появляются объекты посложнее — круги и другие формы. Чем глубже слой сети, тем более сложные признаки она извлекает. На последних слоях нейросеть выявляет признаки и пытается соотнести их с конкретными изображениями и определить, что именно на них нарисовано. Сложные признаки, которые получились на последних слоях, нейросеть будет классифицировать с помощью полносвязного, чтобы найти ответ на вопрос, что изображено на картинке, человек или что-либо ещё.

Существует множество моделей свёрточных нейронных сетей. В дипломной работе рассматриваются такие как:

- VGG-16,
- MobileNet,
- MobileNetV2.

«VGG-16» — модель свёрточной нейронной сети, которая была предложена учеными из Оксфордского университета. Описаны основные достоинства и недостатки модели VGG-16 (например, модель достигает точности 92.7%, однако она имеет очень медленную скорость обучения, и архитектура сети весит слишком много).

«MobileNet» — это архитектура CNN, разработанная исследователями Google в 2017 году и используемая для эффективного внедрения Computer Vision в небольшие портативные устройства, такие как мобильные телефоны и роботы, без значительного снижения точности. Общее количество параметров в стандартной MobileNet составляет 4,2 миллиона, что значительно меньше, чем у некоторых других архитектур CNN. MobileNets также дает разработчикам моделей возможность гибко контролировать размер своей модели (за счет точности) в зависимости от своих требований. путем введения двух новых глобальных гиперпараметров, которые можно настроить в соответствии с требованиями разработчика модели.

«MobileNetV2» — следующее поколение нейронных сетей семейства MobileNet, которое позволяет достигать примерно такой же точности распознавания при

ещё большей скорости работы. Основной строительный блок этой сети в целом похож на предыдущее поколение, но имеет ряд ключевых особенностей. Как и в MobileNet, здесь есть сверточные блоки, однако в новой версии количество блоков — 53. С MobileNetV2 мобильные разработчики получили почти неограниченный инструментарий в области компьютерного зрения — помимо относительно простых моделей для классификации изображений, теперь возможно использовать прямо на мобильном устройстве алгоритмы детектирования объектов и семантической сегментации.

В этом разделе так же говорится о ежегодном мировом соревновании The ImageNet Large Scale Visual Recognition Challenge, которое ежегодно с 2010 до 2017 года, в нем принимали участие более пятидесяти организаций. На нём исследователи сравнивали свои нейронные сети в задачах обнаружения и классификации объектов на фотографиях. ILSVRC использует набор данных, являющийся частью ImageNet. Этот набор данных охватывает 1000 классов объектов и содержит 1 281 167 обучающих изображений, 50 000 проверочных и 100000 тестовых изображений.

Подраздел 1.2 «Инструменты для создания мобильного приложения» рассматривает различные виды инструментов, как для работы с нейронными сетями, так и для создания приложения на ОС Android. Также описаны поставленные задачи создания мобильного приложения по классификации змей. Основными инструментами для создания мобильного приложения стали: TensorFlow Lite, Keras, Transfer Learning, Android Studio.

TensorFlow Lite. Библиотека специально разработана для работы с мобильными устройствами. Обеспечивает высокую производительность и компактный размер моделей. Модель TF lite — это модель специального формата, эффективная с точки зрения точности, а также облегченная версия, которая будет занимать меньше места. Эти особенности делают модели TF Lite подходящими для работы на мобильных и встраиваемых устройствах.

TensorFlow Lite предоставляет интерфейс для использования аппаратного ускорения, если оно доступно на устройстве. Это делается с помощью библиотеки Android Neural Networks, выпущенной как часть Android O-MR1.

Keras — открытая библиотека, написанная на языке Python и обеспечивающая взаимодействие с искусственными нейронными сетями. До версии 2.3 (2019) поддерживались разные нейросетевые библиотеки — как TensorFlow,

Microsoft Cognitive Toolkit, DeepLearning4j, и Theano, впоследствии сохранена только поддержка TensorFlow. Нацелена на оперативную работу с сетями глубинного обучения, при этом спроектирована так, чтобы быть компактной, модульной и расширяемой. Библиотека Keras обладает богатым набором инструментов для обучения нейронных сетей и поддерживают экспорт моделей в формат TensorFlow Lite.

Transfer Learning (трансферное обучение) — это подраздел машинного обучения, целью которого является применение знаний, полученные из одной задачи, к другой целевой задаче.

2 аспекта при использовании Transfer Learning:

1. Наличие открытой предварительно обученной модели;
2. Переназначение модели с помощью выделения признаков (Feature extraction) или дообучения (Fine Tuning).

Предварительно обученная модель — это модель, созданная и обученная кем-то другим для решения проблемы, аналогичной нашей. На практике кто-то, имея большие вычислительные ресурсы, они конструируют большую нейронную сеть для решения конкретной проблемы, тренируют её на большом наборе данных (Big Data), например ImageNet или Wikipedia Corpus. Так, например, VGG19 имеет 143667240 параметров и используется для классификации изображений. Под «открытой» моделью мы подразумеваем, что она обнародована, и её можно свободно использовать.

В архитектурах Глубокого обучения начальные слои изучают общую информацию, а слои на последнем уровне более специфичные признаки. Например, первые слои запоминают текстуру, цвет, общую картину, а последние слои глаза, рот, родинки и т.д. Многие модели обучены на всевозможных ситуациях, например, ImageNet содержит 1 миллион изображений с 1000 классами, поэтому нет необходимости изменять общую картину, которую видит текущая модель. Вместо этого полезнее дополнить новыми конкретными признаками, дообучив только последние слои, чтобы переделать её для собственных нужд.

Android Studio — универсальная среда разработки, так как позволяет оптимизировать работу будущих приложения для работы не только на смартфонах, но и на планшетах, портативных ПК, которые работают на основе рассматриваемой операционной системы.

Второй раздел «Разработка приложения» включает в себя этапы создания

приложения для классификации змей, а именно постановка задачи, обработка данных из выбранного набора, создание и обучение свёрточной нейронной сети, создание мобильного приложения и его тестирование.

В подразделе 2.1 «Постановка задачи» представлен набор данных, состоящих из 35 папок с фотографиями змей, 35 различных видов. Также описана задача — разработать при помощи интегрированной среды разработки Android Studio мобильное приложение, к которому нужно было подключить обученную на взятом наборе данных свёрточную нейронную сеть MobileNet.

Подраздел 2.2 «Обработка данных из выбранного набора» описывает то, как проходила обработка данных. Сначала была произведена инициализация всех необходимых библиотек и функций, которые понадобятся как с данными, так и с нейронной сетью. Далее производится чтение данных из директории и создание dataset, и вывод информации о том, какие классы змей имеются в наборе данных. После этого подготавливаются данные для обучения нейронной сети.

Подраздел 2.3 «Создание и обучение свёрточной нейронной сети» отражает основные аспекты при создании и обучении свёрточной нейронной сети для классификации змей. За основу была взята свёрточная нейронная сеть MobileNet, так как она показывает хорошие результаты точности работы и, как отмечается в подразделе, после сохранения занимает всего 17 мегабайт. Построение модели было выполнено при помощи Transfer Learning, путем замораживания верхних слоев и добавлением слоя пулинга и нескольких полносвязных слоев. Также были представлены результаты обучения модели нейронной сети с точностью 93.61% и 83.81% на тестовой выборке.

В подразделе 2.4 «Создание мобильного приложения» описываются этапы создания данного приложения. Уточняется, что мобильное приложение было написано при помощи интегрированной среды разработки Android Studio, которая в свою очередь работает на языке программирования Java. Был создан новый проект и указано его название. Далее создавалась внешняя оболочка приложения, состоящая из 2 кнопок, 2 текстовых полей для вывода информации о змеях и одно поле для вывода картинки. Потом была добавлена нейронная сеть, скомпилированная в расширение .tflite. Далее были добавлены две функции — в первой функции при нажатии на кнопку камеры сначала выполняется проверка на системное разрешение пользоваться камерой, если оно

есть, то камера запустится; вторая функция при нажатии на кнопку галереи предоставит нам всплывающее окно на доступ в галерею. Также создаются функции, для проверки разрешения на создание фото или доступа в галерею. И описывается функция классификации, которая на вход получает растровое изображение, а на выход выдает результат в виде названия класса змеи. Для этого копируется пример кода для классификации, который был получен при добавлении нейронной сети с расширением `.tflite` в проект. Создается байтовый буфер для хранения картинки в виде числовой информации о каждом пикселе входной картинки, после растровое изображение обрабатывается в цикле по каждому пикселю, и информация добавляется в байтовый буфер, который после этой обработки и заполнения поступает на вход нейронной сети. Она выполняет классификацию, и выдает массив вероятностей совпадения принятой картинки с каждым классом. Далее в цикле находится самая большая вероятность и сопоставляя с номером класса выводится результат на экран в виде текста, а том какой класс у этой змеи и ядовита ли она. Информация о ядовитости змей была добавлена вручную для каждого класса, собрав данные из сети Интернет и добавив эти данные в вывод. Отмечается, что после выполненной работы был собран арк файл, для того, чтобы его можно было установить на смартфон.

Подраздел 2.5 «Тестирование мобильного приложения» показывает, как работа приложения была протестирована на изображениях змей, найденных в интернете.

ЗАКЛЮЧЕНИЕ

В ходе выполнения данной дипломной работы была достигнута поставленная цель, а именно разработано мобильное приложение для классификации змей при помощи сверточной нейронной сети MobileNet. Мобильное приложение позволяет пользователю быстро и точно определять вид змеи на изображении.

Так же выполнены поставленные задачи:

- Рассмотрены нейронные сети предрасположенные для классификации изображений;
- Изучены инструменты для создания мобильных приложений для ОС ANDROID;
- Обучена нейронную сеть;
- Разработано мобильное приложение на основе обученной нейронной сети.

В результате работы было создано полноценное мобильное приложение, которое может быть использовано для классификации змей на фотографиях. Однако, возможности приложения не ограничиваются только этим. В дальнейшем можно рассмотреть возможность расширения базы данных для распознавания других видов животных или добавление функционала для обучения нейронной сети на пользовательских данных.

Таким образом, данная работа имеет практическую значимость и может быть использована в различных областях, связанных с распознаванием объектов на изображениях при помощи нейронных сетей и машинного обучения.