

УДК

ПРИМЕНЕНИЕ КОМПЬЮТЕРНОГО ЗРЕНИЯ И MLOPS В СФЕРЕ МЕДИЦИНСКОГО ОБСЛУЖИВАНИЯ

Шкарубский М.А. студент группы 153503, Владимирцев В. Д., ассистент каф. Информатики

*Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь*

Владимирцев В. Д. – ассистент кафедры информатики

Аннотация. В данной статье описывается применение современных технологий в сфере машинного обучения, больших данных и MLOps в медицинской сфере. Значительное внимание уделяется компьютерному зрению и возможным сценариям его применения. В работе также рассматривается использование облачных сервисов для разворачивания моделей и налаживания процессов, с целью создания надежной, расширяемой системы, способной не только давать точные прогнозы, но и надежно работать практически при любых обстоятельствах.

Ключевые слова. Машинное обучение (Machine Learning), MLOps, Компьютерное Зрение (Computer Vision), Большие Данные (Big Data), Google Cloud Platform, API, DICOM, медицина.

На сегодняшний день, современные технологии машинного обучения (ML) находят свое применение в самых разных сферах деятельности человека. От рекомендательных систем в социальных сетях и музыкальных сервисах, до анализа речи и эмоций человека – область применения алгоритмов ML действительно широка. Эта статья выделяет одно из главных направлений ML – компьютерное зрение, и рассказывает о том, как оно может улучшить медицинскую сферу уже сегодня.

Компьютерное зрение - это важная технология машинного обучения, которая находит широкое применение в медицине. Оно используется для анализа медицинских изображений, таких как РЭМ и МРТ. При этом компьютерное зрение может обнаруживать даже те признаки заболевания, которые не могут быть замечены человеческим глазом, что позволяет существенно улучшить точность диагностики и эффективность лечения. Более того, компьютерное зрение может быть использовано для анализа других типов медицинских изображений, что может помочь в различных областях, например, в планировании хирургических вмешательств, в оценке состояния костной ткани, или нахождении раковых опухолей в человеческом мозгу.

Однако, ни одна система, применяющаяся в промышленном масштабе, не способна существовать без четко налаженных процессов. Такая система не будет расширяемой, легко поддерживаемой и актуализируемой, а ведь от этого напрямую зависит ее эффективность. Часто бывает, что распределение входных данных, как и целевых переменных со временем может изменяться, а модель (или совокупность моделей), должна отвечать этим изменениям, и подстраиваясь под них, и в то же время не терять свою обобщающую способность. Другим также вполне реальным сценарием является переход на новые технологии: к примеру, команда ML-инженеров решила, что разумным будет попробовать использовать ResNet вместо Inception [1] в рамках текущей задачи. Для совершения такого перехода нужно иметь гибкую и легко тестируемую систему, чтобы проверить эффективность новой нейронной сети, провести сравнительный анализ и без больших задержек заменить старую модель новой. Здесь и приходит на помощь MLOps.

MLOps - это методология, которая объединяет несколько этапов в жизненном цикле моделей машинного обучения. Она включает в себя процессы разработки, тестирования, внедрения и управления моделями машинного обучения, а также управления данными. Важным аспектом MLOps является автоматизация процессов. Автоматизация позволяет сократить время, затрачиваемое на каждый этап, приводит к меньшей вероятности ошибок и упрощает масштабирование. MLOps также включает в себя мониторинг и управление моделями после их внедрения. Это важный аспект, который позволяет быстро обнаруживать проблемы и улучшать модели в реальном времени. Кроме вышеупомянутых плюсов, MLOps может быть использована для управления версиями моделей и данных, что облегчает совместную работу нескольких команд и ускоряет процесс разработки. Пример CI/CD и автоматизированного процесса разработки ML продукта показан на рисунке 1.

На данный момент существует несколько основных платформ, позволяющих относительно легко и быстро разворачивать целые приложения с использованием MLOps. Одним из примеров таких платформ является Google Cloud Platform (GCP) [2], которая имеет ряд необходимых платформ и API для взаимодействия с ними. Эти сервисы позволяют создавать, обучать и запускать

модели машинного обучения, а также обрабатывать большие объемы данных. Например, Vertex AI – это платформа, которая позволяет разрабатывать, обучать и запускать модели машинного обучения. Она обеспечивает автоматическое масштабирование и оптимизацию моделей, а также позволяет управлять их жизненным циклом. Еще одним важным сервисом является Cloud Storage [3] – сервис, который позволяет хранить и управлять данными. Он обеспечивает высокий уровень безопасности и доступности данных, а также позволяет легко интегрировать данные с другими сервисами Google Cloud Platform.

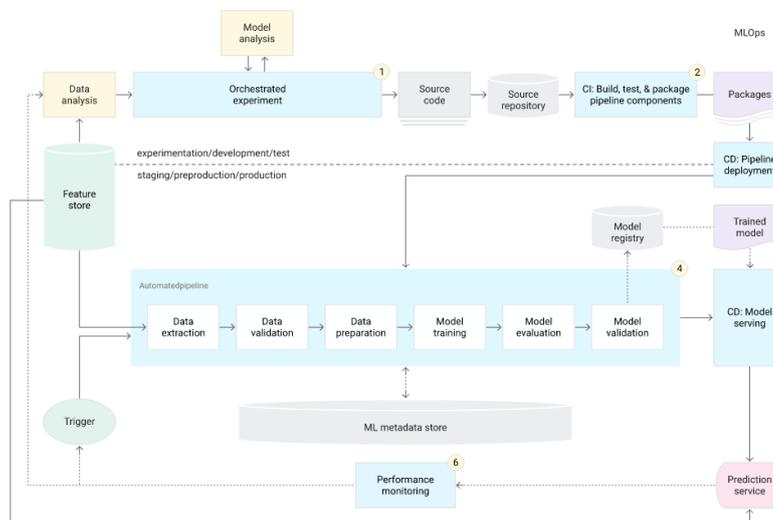


Рисунок 1 – автоматизация всего ML пайплайна с использованием continuous integration и continuous delivery.

Все вышеописанные субплатформы GCP, как и само машинное обучение (а именно нейронные сети, являющие собой основу компьютерного зрения), нашли свое применение в рамках курсовой работы по предмету «Объектно-ориентированное программирование». Суть работы состоит в разработке сервиса, помогающего врачам делать прогнозы о наличии у пациента раковой опухоли в мозгу на основе 3D скана его мозга. Обучение модели (как, впрочем, и весь проект) базируется на датасете, предоставленном некоммерческой организацией «Радиологическое общество Северной Америки» на ресурсе Kaggle в рамках одного из соревнований [4]. Датасет состоит из файлов в формате .dcm, каждый из которых соответствует срезу одного из четырех форматов: T1wCE, T1w, T2w, FLAIR, которые позволяют исследовать мозг на наличие в нем глиобластомы – самой частой и агрессивной формы опухоли мозга, которая составляет до 52 % первичных опухолей мозга и до 20 % всех внутримозговых опухолей [5]. Структура данных соревнования определяется тремя частями: Обучение, Валидация (Публичная) и Тестирование (Приватная). «Обучающая» и «Валидационная» части предоставляются участникам, в то время как часть для тестирования остается скрытой на протяжении всего соревнования и после его окончания. Эти 3 части организованы следующим образом: каждый независимый случай (в сущность – анонимизированный пациент) имеет отдельную папку, идентифицируемую пятизначным номером. В каждой из этих папок есть четыре подпапки, каждая из которых соответствует каждому из структурных MPT-сканов (mpMRI).

Задача создания такого сервиса делится на несколько основных частей, каждая из которых также может включать в себя некоторые подзадачи: исследовательский анализ данных (EDA), выбор архитектуры и построение модели, развертывание модели в облачном сервисе.

Главной целью проведения EDA в данном проекте являлась выработка понимания формата DICOM, поиск особенностей сканов мозга различных типов, сравнение сканов больных людей со сканами здоровых и оценка распределения классов (целевой переменной). Для этого в проекте использовалась среда Jupyter Notebook, позволяющая точно манипулировать блоками кода, а не запускать всю программу целиком. Помимо этого, EDA помог обнаружить и исправить существенную проблему – неупорядоченность хранения данных в пределах одного 3D скана исходном датасете. Такая неупорядоченность могла навредить модели на этапе обучения, так как топология данных теряется, что приводит к потере эффективности свертки – признаки, которые она могла выделить на этапе обучения нарушены по умолчанию. Решением проблемы являлся переход от EDA к разработке программного продукта.

Для более надежной работы с данными было принято решение ввести две сущности – ScanSlice и Scan3D, представляющих собой двумерный скан (срез), и трехмерный скан, то есть совокупность срезов, соответственно. Каждый экземпляр класса Scan3D сортировал хранящиеся в

нем срезы, используя в качестве ключа сортировки метаданные среза, а именно поле Instance Number, которое отвечало за позицию (в контексте очередности) среза. Сравнение сортированных и несортированных срезов проводится на рисунке 2. Еще одной довольно серьезной проблемой являлось неодинаковое количество срезов (фактически глубина) каждого трехмерного скана. Довольно популярной практикой для ее решения было выделение центральной части изображения фиксированной глубины, что приводило к тому, что данные теряли свою репрезентативность. В рамках курсовой работы было принято альтернативное решение: было решено дополнять все сканы пустыми срезами до определенного, общего для всех сканов размера, после чего обучать модель уже на новых данных.

После формализации представления данных в коде и дальнейшего исследования было принято решение использовать модель InceptionV3 на первых этапах разработки продукта. Она отличается большим количеством слоев, но относительно небольшим (например, относительно моделей класса VGG) числом параметров, благодаря свёрточным слоям 1×1 (1×1×1 в случае трехмерной свертки), понижающим размерность входов последующих слоев. Хорошей практикой является использование вначале не самой сложной с точки зрения архитектуры и количества слоев модели и попытка подбора таких валидационных стратегий, которые действительно отражают поведение модели при обучении на данных (подтверждением тому является использование этой практики лидерами соревнования).

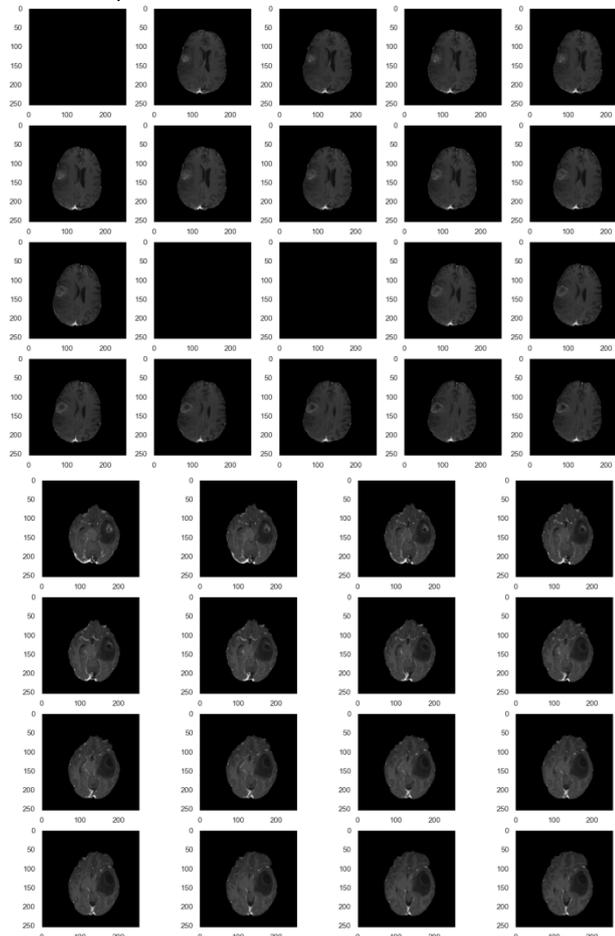


Рисунок 2 – слева находятся неотсортированные срезы в формате T1wCE пациента №43, справа – сортированные. Можно заметить что слева есть пустые изображения, что говорит о том, что срезы, которые на них изображены, пусты, и взяты либо сверху, либо снизу, но никак не с середины. Справа же имеем последовательную картину, где опухоль постепенно увеличивается.

Для имплементации модели, можно воспользоваться Vertex AI SDK, являющимся более высокоуровневой версией Vertex AI Client library. SDK предоставляет возможность создавать датасеты, манипулировать экспериментами (Experiments) и работами (Jobs). Для хранения тренировочных данных разумно использовать Google Cloud Storage, данные из которого можно напрямую направлять в датасеты Vertex AI, в том числе и мини-батчами.

Чтобы обучить модель на тренировочных данных, можно воспользоваться встроенным сервисом AutoML, который автоматически создаст работы, настроит гиперпараметры и оценит качество работы модели с помощью кросс-валидации. Для более гибкой работы имеет смысл создавать свои работы, используя либо Python Distributions, либо Docker контейнеры. В рамках

работы можно использовать собственные метрики качества, функции ошибки, тренировочные циклы, и даже способы хранения и подачи данных модель (это особенно важно в контексте работы с большими данными, т.к. такие данные труднее обрабатывать, что подразумевает использование более подходящих способов хранения и манипуляции, таких как TFRecord, например), что делает Vertex AI крайне мощной платформой, предоставляющих целый ряд решений для одной задачи. Для более продвинутой работы с Vertex AI следует обратиться к документации GCP [5].

Таким образом, данная работа описывает MLOps и ее важность для промышленной реализации алгоритмов машинного обучения. В документе представлен обзор платформ и API для MLOps, с акцентом на Google Cloud Platform, а также пример сервиса, разработанного для прогнозирования опухолей головного мозга с использованием 3D сканов головного мозга и различных сервисов Google Cloud Platform.

Список использованных источников:

1. *MLOps: Continuous delivery and automation pipelines in machine learning* – [Электронный ресурс] Электронные данные. – Режим доступа: Google Clout Platform (<https://cloud.google.com/architecture/mlops-continuous-delivery-and-automation-pipelines-in-machine-learning>).
2. *Cloud Storage documentation* – [Электронный ресурс] Электронные данные. – Режим доступа: Google Cloud Platform (<https://cloud.google.com/storage/docs>).
4. U.Baid, et al., "The RSNA-ASNR-MICCAI BraTS 2021 Benchmark on Brain Tumor Segmentation and Radiogenomic Classification", arXiv:2107.02314, 2021.
5. Глиобластома – [Электронный ресурс] Электронные данные. – Режим доступа: Wikipedia.org (<https://www.wikipedia.com/ru/Глиобластома>).
6. *Vertex AI documentation* – [Электронный ресурс] Электронные данные. – Режим доступа: Google Cloud Platform (<https://cloud.google.com/vertex-ai/docs>).
7. *Going deeper with convolutions* / Christian Szegedy [et al.] // Cornell University, 2014.

UDC

APPLICATION OF COMPUTER VISION AND MLOPS IN THE FIELD OF MEDICAL CARE

Shkarubski M. A.

Belarusian State University of Informatics and Radioelectronics, Minsk, Republic of Belarus

Vladymtsev V.D. – Assistant of the Department of Informatics

Annotation. This article describes the application of modern technologies in the field of machine learning, big data and MLOps in the medical field. Considerable attention is paid to computer vision and possible scenarios of its application. The paper also discusses the use of cloud services for deploying models and establishing processes, in order to create a reliable, extensible system that can not only give accurate forecasts, but also work reliably under almost any circumstances.

Keywords. Machine Learning, Apps, Computer Vision, Big Data, Google Cloud Platform, API, DICOM, Medicine.