Управление моделями машинного обучения (ML-model management)

Разделы: Бизнес-задачи

Так же как и любой вид программного обеспечения, реализующего определенные функции, аналитические модели, основанные на машинном обучении (ML-модели), имеют свой жизненный цикл — интервал времени между моментом принятия решения о необходимости ее создания и моментом полного изъятия модели из эксплуатации.

Комплекс организационных и технических мероприятий, направленных на поддержку процессов жизненного цикла ML-модели, называется **управление моделями**. Оно является фундаментальной частью любого конвейера анализа данных с использованием машинного обучения.

Не следует путать понятия **управление жизненным циклом (product lifecycle managemetn — PLM)** и **управление моделями (model management)**. Предметная область PLM — управление организацией, а управление моделями лежит в сфере науки о данных, хотя модель тоже может рассматриваться как продукт.

Необходимость выделения управления ML-моделями в отдельную бизнес-функцию, отличную от управления программным обеспечением вообще, обусловлена наличием интеллектуальной составляющей в их работе, а также важностью полученных результатов, от которых зависит нормальное функционирование бизнеса в целом. В то же время проблемы управления другими видами программных средств, например CRM, носят более локальный характер, затрагивая только некоторые направления бизнеса.

ML-модель — это особый тип программного обеспечения. В обычных программах алгоритм представляет собой жестко закодированный набор инструкций для вычисления детерминированного ответа. В то же время в ML-моделях инструкции, на основе которых формируется ответ, автоматически выводятся из набора данных. Поэтому отношение к управлению ML-моделями как к управлению обычным программным обеспечением является в корне ошибочным, но тем не менее характерным для многих компаний.

Ведь именно благодаря моделям данные становятся источником аналитических отчетов, бизнес-правил, прогнозов, рекомендаций и решений, повышающих эффективность бизнеса. И если данные часто называют «новой нефтью», то модели можно сравнить с насосами, которые приводят в движение потоки данных и извлекают из них знания для бизнеса. Именно поэтому системы управления моделями являются столь важным: без них аналитикам данных было бы очень сложно создавать, отслеживать, сравнивать и развертывать модели.

На практике, в бизнесе, управляемом данными , одновременно может использоваться множество аналитических ML-моделей, которые взаимодействуют друг с другом. В этом случае необходима выработка политики управления моделями, которая полностью удовлетворяет всем потребностям бизнеса. Все действия по поддержке процессов жизненного цикла ML-модели (например, выбор конфигурации, обучение, управление версиями и т.д.) должны быть подчинены политике управления моделями.

Рассмотрим базовые компоненты рабочего процесса управления моделями.

1. Изучение предметной области и бизнес-процессов, где будет работать модель, принятие решения о целесообразности ее создания. Разработка концепции использования модели, формулирование целей и задач ее применения. Выдвижение гипотез, которая должна будет подтвердить или опровергнуть модель.
2. Разработка и создание инфраструктуры данных для модели. Выбор источников данных (хранилище данных, файловое хранилище, локальные источники и т.д.), разработка и реализация процессов ETL или ELT (при необходимости), формирование обучающей, тестовой и валидационной выборок и оценка их репрезентативности. Целесообразно создание единого источника данных.
3. Выбор типа модели, ее архитектуры и конфигурации. Тип модели (нейросеть, дерево решений, k-средних, линейная регрессия и т.д.) определяется задачей, которую должна решать модель (классификация, кластеризация, прогнозирование и т.д.), а также особенностями данных. Архитектура модели определяет общие принципы ее структуры (например, плоскослоистая нейронная сеть или полносвязная). Конфигурация — это конкретная реализация модели (число слоев в нейронной сети, количество нейронов в слоях и т.д.). Как правило, выбор оптимальной архитектуры и конфигурации является результатом многократных экспериментов, что порождает большое количество конфигураций и версий, которые требуется хранить, анализировать и сопоставлять.
4. Разработать и реализовать стратегию очистки и предобработки данных, которая будет индивидуальна для конкретной модели, потому что одни модели устойчивы к наличию шумов, пропусков и выбросов в данных, а другие чувствительны. Подбор используемых алгоритмов и параметров очистки и предобработки также обычно является результатом многочисленных экспериментов.
5. Обучение модели заключается в подстройке (обычно итеративной) гиперпараметров модели на основе выбранного алгоритма обучения с заданными параметрами. Выбор самого алгоритма и его параметров также настраивается экспериментально. При этом могут использоваться различные функции потерь.
6. Оценка результатов обучения модели — точности и обобщающей способности.
7. Передача модели в эксплуатацию и ее развертывание в бизнес-среде. Производится синхронизация модели с бизнес-средой — проверяется, также хорошо модель работает с реальными данными, как и с обучающими. Если это не так, то может производиться коррекция модели (дообучение). Одной из причин необходимости такой синхронизации является утечка данных. Рекомендуется предусмотреть автоматический откат, если при вводе модели в бизнес-процесс что-то пошло не так.
8. Мониторинг качества работы модели с целью своевременного выявления его падения ниже критического уровня по причине деградации модели из-за утечки или дрейфа данных. Если это произошло, производится повторное обучение модели.

9. Вывод модели из эксплуатации. Обычно производится в двух случаях. Первый — когда бизнес-процесс, связанный с моделью, завершается (например, прекращается выпуск товара, спрос на который прогнозировала модель). Другой случай, когда деградация модели из-за дрейфа данных (или иных факторов) зашла так далеко, что восстановить работоспособность модели путем коррекции не представляется возможным, и приходится заменять ее на новую. При выводе модели из эксплуатации должны учитываться ее возможные связи с другими моделями, чтобы не нарушилась их нормальная работа. Кроме этого полезно будет задокументировать и сохранить результаты всех экспериментов, параметров модели и их изменений в процессе ее эксплуатации, чтобы в дальнейшем их можно было использовать при решении аналогичных задач.

Приведенная структура является достаточно общей. При решении конкретных задач в процессе управления ML-моделями могут появляться другие этапы, а некоторые из указанных — не использоваться. Например, если обучающие данные уже имеют достаточно высокий уровень качества, то этап очистки и предобработки может быть опущен, что случается, впрочем, весьма редко.

Для комплексной поддержки процессов жизненного цикла модели в конвейер анализа данных могут встраиваться следующие компоненты:

- **Система управления версиями и конфигурациями**. Позволяет регистрировать, хранить, анализировать и выбирать все изменения в состоянии модели, которые происходили в ее жизненном цикле.
- **Локатор экспериментов**. Отслеживает, собирает и упорядочивает данные об обучении и тестировании производительности модели в течение множества прогонов с различными конфигурациями и наборами данных.
- **Реестр моделей**. В случае, если в конвейере анализа данных используется множество моделей, то этот компонент позволяет отслеживать их текущее состояние: какие модели обучаются или тестируются, а какие уже развернуты и эксплуатируются.
- **Дашборды**. Информационные панели с различными визуальными компонентами для мониторинга процессов обучения и эксплуатации моделей.

Комплексная организация управления моделями в аналитических проектах ведет за собой дополнительные временные и материальные затраты. Тем не менее, во многих случаях они позволяют избежать потерь, связанных с плохой работой аналитических ML-моделей, и поэтому оправданны.