СЕКЦИЯ 2
СИСТЕМЫ АВТОМАТИЗАЦИИ ПРОЕКТИРОВАНИЯ

DOI: 10.58168/IntInfSys2026_51-58
УДК 004.85

КОНВЕЙЕР РАЗВЁРТЫВАНИЯ И МОНИТОРИНГА МОДЕЛЕЙ МАШИННОГО ОБУЧЕНИЯ С КОНТРОЛЕМ ДРЕЙФА ДАННЫХ

В.Э. Буздин[1], Е.А. Аникеев[1]

[1]ФГБОУ ВО «Воронежский государственный лесотехнический университет имени Г.Ф. Морозова»

Аннотация. Рассмотрен конвейер вывода моделей машинного обучения в эксплуатацию, объединяющий автоматическое развёртывание, мониторинг качества и контроль дрейфа входных данных. Предложена схема с обратной связью, в которой переобучение запускается по индексу дрейфа. На учебном примере показано, что контроль дрейфа сокращает время обнаружения деградации модели и простой на устаревшей версии без постоянного переобучения по расписанию.

Ключевые слова: машинное обучение, развёртывание моделей, дрейф данных, мониторинг качества, переобучение, конвейер, обработка информации, обратная связь, интеллектуальные системы.

A PIPELINE FOR DEPLOYING AND MONITORING MACHINE LEARNING MODELS WITH DATA DRIFT CONTROL

V.E. Buzdin[1], E.A. Anikeev[1]

[1]Voronezh State University of Forestry and Technologies named after G.F. Morozov

Abstract. A pipeline for putting machine learning models into operation is considered, combining automatic deployment, quality monitoring, and control of input data drift. A feedback scheme is proposed in which retraining is triggered by a drift index. Using an educational example, it is shown that drift control reduces the time to detect model degradation and the downtime on an outdated version without constant scheduled retraining.

© Буздин В. Э., Аникеев Е. А.

Keywords: machine learning, model deployment, data drift, quality monitoring, retraining, pipeline, information processing, feedback, intelligent systems.

Интеллектуальные информационные системы всё чаще включают модели машинного обучения: классификаторы, прогнозы спроса, обнаружение мошенничества, рекомендации. В отличие от обычного программного модуля, обученная модель имеет особенность – её качество со временем падает, даже если код не менялся. Причина в том, что меняются сами данные: пользователи ведут себя иначе, появляются новые категории, смещаются распределения признаков. Это явление называют дрейфом данных, и из-за него модель, точная на момент обучения, постепенно начинает ошибаться.

Поэтому вывод модели в эксплуатацию нельзя сводить к разовому развёртыванию. Нужен повторяемый процесс, который не только доставляет модель в работающую систему, но и следит за её качеством и за изменением входных данных, а при необходимости запускает переобучение. Такой процесс по аналогии с практиками автоматизации поставки программного обеспечения [1] выстраивают как конвейер с обратной связью.

На практике встречаются две крайности. В первой модель обучают один раз и забывают – тогда деградация обнаруживается поздно, по жалобам пользователей. Во второй модель переобучают по расписанию (например, еженедельно) независимо от того, изменились данные или нет, это расходует ресурсы и иногда даже вредит, если свежих размеченных данных мало. Разумная середина – переобучать тогда, когда это действительно нужно, то есть по факту дрейфа.

Цель работы – описать конвейер развёртывания и мониторинга модели машинного обучения, в котором переобучение запускается по измеримому признаку дрейфа входных данных, и оценить, что это даёт по сравнению с переобучением по расписанию и с отсутствием мониторинга.

Рассмотрим типовую задачу: модель-классификатор используется в информационной системе и обрабатывает поток заявок. У модели есть два рода показателей. Первый – качество предсказаний (например, доля верных решений или площадь под ROC-кривой, ROC-AUC); его можно посчитать только тогда, когда становится известен истинный ответ, а это бывает с задержкой. Второй – характеристики входных данных (распределения признаков); их видно сразу, ещё до того, как известен правильный ответ. Идея мониторинга дрейфа в том,

чтобы по второму, доступному немедленно, показателю заранее предупреждать о вероятном падении первого.

Контроль качества модели перед вводом в эксплуатацию и в ходе работы по смыслу близок к тестированию и контролю качества программного обеспечения [2]: прежде чем заменить рабочую версию, новая модель должна пройти проверку на отложенной выборке и не уступить текущей. Сам процесс удобно описывать как систему с обратной связью, где наблюдаемое состояние модели влияет на решение о её обновлении [3].

Предлагаемый конвейер показан на рисунке 1. Он состоит из пяти стадий: подготовка данных и признаков, обучение модели, валидация качества, развёртывание модели как сервиса (serving) и мониторинг качества и дрейфа. Принципиальным элементом является обратная связь: стадия мониторинга при обнаружении значимого дрейфа запускает переобучение, и цикл повторяется уже на свежих данных.

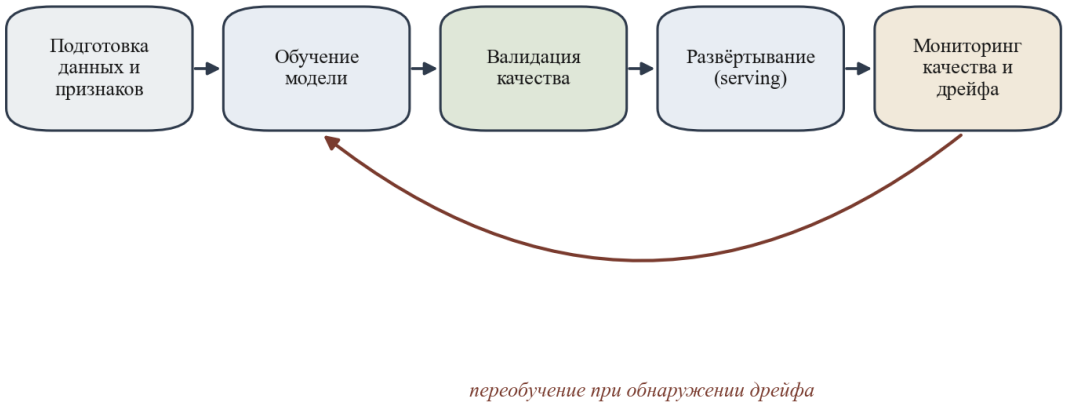


Рисунок 1 – Конвейер вывода модели в эксплуатацию с переобучением по обратной связи

Стадия валидации играет роль защитного барьера: новая модель выводится в эксплуатацию только если на отложенной выборке она не хуже текущей. Развёрнутая модель работает как сервис, к которому обращается остальная система; при росте потока запросов сервис масштабируется горизонтально, а распределение запросов между его экземплярами решается средствами балансировки нагрузки [4]. Стадия мониторинга собирает два потока сведений – отложенные

оценки качества и характеристики входных данных – и сравнивает их с эталонными значениями, полученными при обучении.

Такое построение делает поведение системы предсказуемым и объяснимым: каждое переобучение имеет причину (зафиксированный дрейф), а каждая замена модели проходит через валидацию. По духу это близко к интеллектуальным системам управления качеством, где решение принимается не вручную, а по измеренным показателям процесса [5].

Для числовой оценки дрейфа применялся индекс стабильности популяции (population stability index, PSI). Область значений каждого признака разбивается на интервалы; для эталонной (обучающей) выборки и для текущего потока считаются доли наблюдений, попавших в каждый интервал, после чего вычисляется суммарное расхождение этих долей:

$$PSI = \sum_i (q_i - p_i) \cdot \ln(q_i / p_i) \qquad (1)$$

где p_i – доля наблюдений в i-м интервале для эталонной выборки;

q_i – доля для текущего потока.

Чем сильнее текущее распределение отличается от эталонного, тем больше значение по формуле (1). Приняты привычные границы интерпретации: PSI до 0,1 – дрейфа практически нет; от 0,1 до 0,2 – умеренный дрейф, требующий внимания; свыше 0,2 – значимый дрейф, при котором назначается переобучение. Индекс рассчитывается отдельно по каждому значимому признаку, а в качестве сигнала берётся наибольшее из значений.

Достоинство такого показателя в том, что он не требует знания истинных ответов и доступен сразу. Это и позволяет предупреждать деградацию заранее: рост PSI обычно опережает падение качества, давая время подготовить и проверить новую модель до того, как ошибки станут заметны пользователям.

Конвейер проверялся на учебной задаче классификации заявок с потоком данных за 24 недели. Эталонные распределения признаков и исходное качество модели зафиксированы на момент обучения (ROC-AUC около 0,92). Затем по неделям отслеживались качество модели и индекс дрейфа PSI. Результаты показаны на рисунке 2.

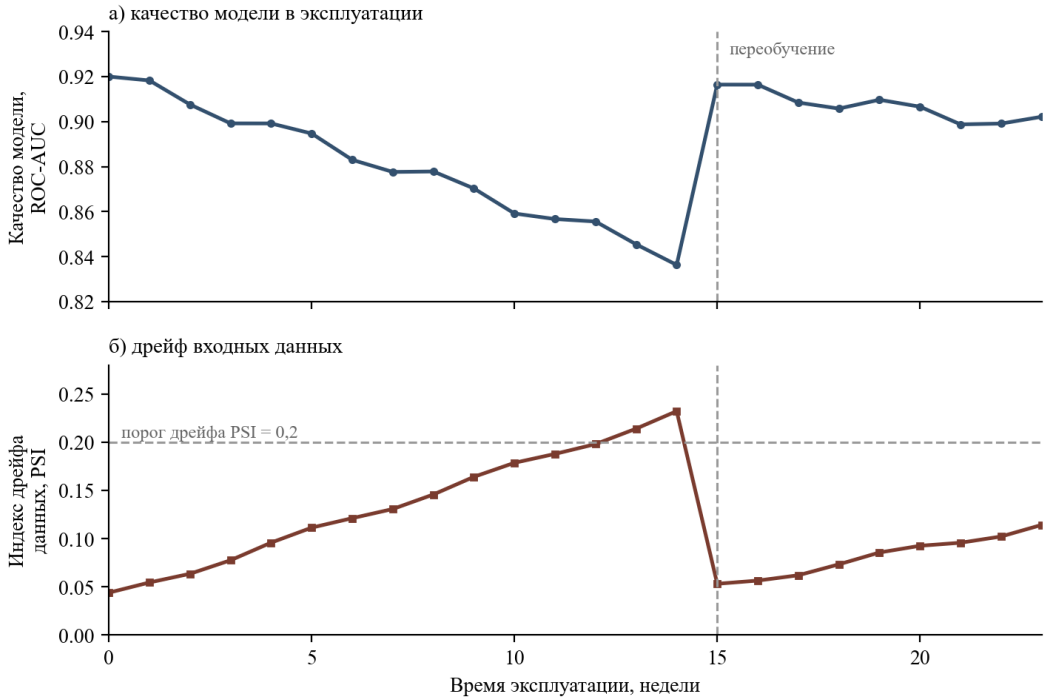


Рисунок 2 – Качество модели (а) и индекс дрейфа входных данных (б) в ходе эксплуатации

Видно, что по мере накопления изменений во входных данных индекс PSI рос и к 13–14-й неделе превысил порог 0,2, а качество модели к этому времени снизилось с 0,92 до примерно 0,84. Превышение порога запустило переобучение на свежих данных; после замены модели на 15-й неделе качество вернулось к уровню около 0,91, а индекс дрейфа сбросился, поскольку эталонные распределения были пересчитаны. Важно, что сигнал о дрейфе появился раньше, чем падение качества стало критичным, – это и есть практическая польза опережающего мониторинга.

Конвейер сопоставлялся с двумя базовыми стратегиями на том же потоке данных: без мониторинга (модель не обновляется) и с переобучением строго по расписанию (раз в 4 недели). Сравнение приведено в таблице 1.

Таблица 1 – Сравнение стратегий сопровождения модели

Стратегия	Мин. ROC-AUC	Переобучений	Простой на плохой модели, нед.
Без мониторинга	0,84	0	10
По расписанию (4 нед.)	0,88	6	3
По дрейфу (PSI)	0,89	2	1

Стратегия без мониторинга показала наибольшее и самое долгое падение качества. Переобучение по расписанию держало качество выше, но потребовало шести переобучений, часть из которых пришлась на периоды без заметного дрейфа и не дала выигрыша. Управление по дрейфу удержало качество на сопоставимом уровне всего двумя переобучениями и с наименьшим простоем на устаревшей модели – ресурсы тратились тогда, когда это было оправдано.

Перевод сопровождения модели в измеримые величины делает решения обоснованными: модель обновляется не «на всякий случай» и не по жалобам, а по зафиксированному признаку дрейфа. Такой подход согласуется с идеей информационно-интеллектуальной поддержки принятия решений, когда выбор момента действия опирается на наблюдаемые показатели, а не на интуицию. Дополнительное преимущество – прослеживаемость: для каждой версии модели известно, на каких данных она обучена и почему была заменена.

Ограничения подхода связаны с тем, что дрейф входных данных не всегда совпадает с падением качества. Возможен дрейф без потери качества (изменение, к которому модель устойчива) и, наоборот, падение качества без заметного дрейфа признаков – например, при изменении самой зависимости между признаками и ответом. Поэтому контроль дрейфа разумно дополнять отложенной оценкой качества, а пороги PSI подбирать под конкретную задачу. Тем не менее даже простой контроль дрейфа заметно улучшает сопровождение модели и не требует сложной инфраструктуры.

Описан конвейер развёртывания и мониторинга модели машинного обучения с обратной связью, в котором переобучение запускается по индексу дрейфа входных данных PSI. На учебном потоке данных за 24 недели показано, что рост дрейфа опережает падение качества и служит ранним сигналом для переобучения. По сравнению с отсутствием мониторинга и переобучением по расписанию управление по дрейфу удержало качество модели меньшим числом переобучений и с наименьшим простоем на устаревшей версии. Предложенный конвейер даёт практичный и объяснимый способ сопровождения моделей в интеллектуальных информационных системах.

Список литературы

1. Баланов, А. Н. DevOps: интеграция и автоматизация: учебное пособие для СПО / А. Н. Баланов. — 2-е изд., стер. — Санкт-Петербург: Лань, 2025. — 68 с. — ISBN 978-5-507-54085-3. — Текст: электронный // Лань: электронно-библиотечная система. — URL: https://e.lanbook.com/book/505387 (дата обращения: 14.06.2026). — Режим доступа: для авториз. пользователей.

2. Турнецкая, Е. Л. Программная инженерия. Тестирование и контроль качества программного обеспечения: учебное пособие для вузов / Е. Л. Турнецкая, А. В. Аграновский. — Санкт-Петербург: Лань, 2025. — 172 с. — ISBN 978-5-507-51677-3. — Текст: электронный // Лань: электронно-библиотечная система. — URL: https://e.lanbook.com/book/455672 (дата обращения: 14.06.2026). — Режим доступа: для авториз. пользователей.

3. Моделирование систем и процессов. Практический курс: учебник для вузов / под редакцией В. Н. Волковой. — Москва: Издательство Юрайт, 2025. — 295 с. — (Высшее образование). — ISBN 978-5-534-01442-6. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/561270 (дата обращения: 14.06.2026).

4. Мочалов, В. П. Алгоритм динамического распределения и балансировки нагрузки в распределенных облачных вычислениях / В. П. Мочалов, Н. Ю. Братченко, Д. В. Гостева // Моделирование систем и процессов. – 2024. – Т. 17, № 1. – С. 92-102. – DOI 10.12737/2219-0767-2024-17-1-92-102. – EDN EWMPYM.

5. Бантюков, С. М. Создание интеллектуальной системы управления качеством предприятия «Эталон» в авиационной промышленности / С. М. Бантюков // Моделирование систем и процессов. – 2024. – Т. 17, № 2. – С. 15-23. – DOI 10.12737/2219-0767-2024-17-2-15-23. – EDN YHXUPY.

References

1. Balanov, A. N. DevOps: integration and automation: a textbook for vocational education / A. N. Balanov. — 2nd ed., ster. — Saint Petersburg: Lan, 2025. — 68 p. — ISBN 978-5-507-54085-3. — Text: electronic // Lan: electronic library system. — URL: https://e.lanbook.com/book/505387 (date of access: 14.06.2026). — Access mode: for authorized users.

2. Turnetskaya, E. L. Software engineering. Software testing and quality control: a textbook for universities / E. L. Turnetskaya, A. V. Agranovsky. — Saint Petersburg:

Lan, 2025. — 172 p. — ISBN 978-5-507-51677-3. — Text: electronic // Lan: electronic library system. — URL: https://e.lanbook.com/book/455672 (date of access: 14.06.2026). — Access mode: for authorized users.

3. Modeling of systems and processes. Practical course: a textbook for universities / edited by V. N. Volkova. — Moscow: Yurait Publishing House, 2025. — 295 p. — (Higher education). — ISBN 978-5-534-01442-6. — Text: electronic // Yurait educational platform [website]. — URL: https://urait.ru/bcode/561270 (date of access: 14.06.2026).

4. Mochalov, V. P. An algorithm for dynamic distribution and load balancing in distributed cloud computing / V. P. Mochalov, N. Yu. Bratchenko, D. V. Gosteva // Modeling of systems and processes. – 2024. – Vol. 17, No. 1. – pp. 92-102. – DOI 10.12737/2219-0767-2024-17-1-92-102. – EDN EWMPYM.

5. Bantyukov, S. M. Creation of an intelligent quality management system of the Etalon enterprise in the aviation industry / S. M. Bantyukov // Modeling of systems and processes. – 2024. – Vol. 17, No. 2. – pp. 15-23. – DOI 10.12737/2219-0767-2024-17-2-15-23. – EDN YHXUPY.