ПОДХОД К ПРИМЕНЕНИЮ МАШИННОГО ОБУЧЕНИЯ В ПРОГНОЗИРОВАНИИ ЗАГРУЗКИ ВИРТУАЛЬНЫХ ВЫЧИСЛИТЕЛЬНЫХ СИСТЕМ

APPROACH OF MACHINE LEARNING APPLICATION IN PREDICTING THE LOAD OF VIRTUAL COMPUTING SYSTEMS

Yu. Nesterov
A. Kalistratov
G. Afanasyev

Summary. The article presents an approach towards predicting the load on the computational systems, expressed in the number of incoming requests for the execution of various tasks, based on the methods of machine learning. By applying the implementation of gradient boosting technology, the authors train the model and then restore the time series with its help. Authors estimate the accuracy and completeness of the obtained forecast. Obtained data can be used to further solution of the problem of forecasting the amount of computing resources required to perform the selected class of tasks.

Нестеров Юрий Григорьевич
К.т.н., доцент, Московский Государственный Технический Университет им. Н.Э. Баумана
ugn@bmstu.ru
Калистратов Алексей Павлович
Аспирант, Московский Государственный Технический Университет им. Н.Э. Баумана
akalistratov@gmail.com
Афанасьев Геннадий Иванович
К.т.н., доцент, Московский Государственный Технический Университет им. Н.Э. Баумана
gaipcs@bmstu.ru

Аннотация. В статье представлен подход к прогнозированию загрузки вычислительных систем, выражающийся в количестве поступающих за отрезок времени заявок на выполнение различного рода задач, основывающийся на методах машинного обучения. Путем применения реализации технологии градиентного бустинга авторы производят обучение модели и последующее восстановление временных рядов с ее помощью. Оцениваются точность и полнота полученного прогноза. Полученные данные могут быть использованы для дальнейшего решения задачи прогнозирования объема вычислительных ресурсов, требуемых для выполнения выбранного класса задач.

Введение

Производственная деятельность любого предприятия в настоящее время невозможна без применения различных информационных систем, функционирование которых обеспечивается вычислительными системами (ВС). В зависимости от масштабов предприятия, требований к коэффициенту готовности и производительности ВС, определяются как количественные требования к вычислительным ресурсам, из которых состоит ВС (количество процессорных ядер, их частота, объем оперативной памяти, объем устройств хранения данных и так далее), так и качественные требования, например, применение виртуализации или использование физических вычислительных ресурсов [1]. В данной статье рассматривается вопрос использования технологий машинного обучения при решении подзадачи прогнозирования загрузки вычислительных систем, которая, в свою очередь, является шагом в решении задачи выбора оптимального набора виртуальных вычислительных систем для решения заданного класса задач. Класс (–ы) задач, решаемых при помощи систем, определяются бизнес-процессами, задействующими эти системы и предметной областью. В качестве предметной области в данной (и других работах авторов) работе выступает процесс разработки программного обеспечения (ПО).

Разработка ПО представляет собой комплекс мероприятий по сбору и разработке требований и преобразованию их в программное обеспечение. Основные этапы разработки ПО состоят из следующих стадий: проектирование, разработка, тестирование, ввод в эксплуатацию, сопровождение. Каждый из этапов разработки характеризуется требуемой для осуществляемой деятельности инфраструктурой, т.е., набором программных и аппаратных средств, позволяющих производить необходимые на этом этапе действия (написание кода, запуск тестов, составление документации, и так далее) [2].

Построение инфраструктуры на основе облачных технологий

Предыдущие этапы решения задачи выбора оптимального набора вычислительных систем привели авторов к выводу, что наиболее рациональной основой построения инфраструктуры, состоящей из вычислительных систем на данный момент, является подход, использующий виртуализацию и облачные вычисления [3].

Облачные вычисления — подход к распределенной обработке данных (вычисление, хранение, преобразование), при котором доступ к серверным мощностям предоставляется как услуга с доступом через интернет или локальную сеть. Данный подход является развитием клиент-серверной архитектуры обработки данных. Под облачными вычислениями обычно понимается предоставление пользователю вычислительных ресурсов в виде услуги. Со стороны потребителя, облачные вычисления — это получение информационных ресурсов в виде услуги у внешнего поставщика, оплата за которую производится в зависимости от объема потребленных ресурсов согласно установленному тарифу.

Таким образом, можно производить оценку стоимости аренды инфраструктуры, необходимой для реализации проекта разработки ПО на основе прогнозирования времени и интенсивности использования выбранной инфраструктуры. Оценка стоимости необходима для анализа прибыльности нынешних и будущих технологических проектов, а значит, и целесообразности инвестиций в разработку. Если для проекта выделено больше ресурсов, чем реально необходимо, такой проект окажется более дорогостоящим, чем должен был быть при грамотной оценке, и приведет к запаздыванию с началом следующего проекта [4].

Прогнозирование нагрузки

Так как задача выбора оптимального набора вычислительных систем решается на основе вводных данных, содержащих в себе загрузку каждой из систем набора, для ее решения необходимо измерить (если речь идет об изменении набора в режиме «реального времени») или спрогнозировать (если речь идет о планировании) эту загрузку. Фактически, эту задачу можно свести к классической задаче прогнозирования временного ряда. Прогнозирование временных рядов является важной научно-технической проблемой, т.к. позволяет предсказать поведение различных факторов в экологических, экономических, социальных и иных системах. Таким образом, основной целью любого прогнозирования является создание некой «машины времени», которая позволяет заглянуть в будущее и оценить тенденции в изменениях того или иного фактора. Такая «машина времени» в большинстве случаев базируется на методах математического моделирования, в частности на построении модельной авторегрессии, скользящей по временному ряду и позволяющей осуществлять экстраполирование на несколько шагов вперед. Качество прогноза в таком случае зависит от наличия предыстории изменяемого фактора, погрешностей измерения рассматриваемой величины, глубины памяти (т.е. числа одновременно учтенных членов временного ряда). [5]

Традиционный подход к прогнозированию включает в себя анализ временного ряда из одного параметра, например, значения оценки, выставленной пользователем [6–7]. Для прогнозирования нагрузки такой подход уместен, так как потребность в вычислительных ресурсах определяется в первую очередь количеством заявок за заданный промежуток времени (обычно час или день). Обычно назначение ресурсов происходит экспертным путем: администраторы вычислительных систем оценивают потребности в ресурсах в реальном времени и на основе своего профессионального опыта принимают решения и назначают те или иные наборы ресурсов виртуальным машинам.

В качестве основного преимущества использования машинного обучения при задаче прогнозирования нагрузки на вычислительные системы можно выделить возможность принятия решений без привлечения экспертов. В данной работе рассматривается несколько параметров, неявно связанных друг с другом: методология разработки, этап разработки, количество пользователей, количество предыдущих итераций. Этот набор можно представить как

$$Project_r = \{\mathrm{Method}_r;\ \mathrm{Stage}_r;\ \mathrm{UserAmount}_r;\ \mathrm{Iteration}_r\}$$

После представления каждого временного отрезка и сопоставления существующих данных о количестве заявок в каждой подсистеме можно составить массив данных, описываемый как

$$Set = \{(Project_r, JobAmount_r)\}_{r=1..n},$$

где $Project_r = \left(Project_r^1, \dots, Project_r^4\right)$ —

вектор из параметров, описывающих взятый отрезок времени для выбранной системы, а $\mathrm{JobAmount}_r$ — количество заявок, поступивших в систему за этот отрезок времени. Задача машинного обучения в этом случае — минимизировать погрешность в результате, выдаваемом моделью относительно реальных значений. Было принято решение использовать модель обучения, основанную на градиентном бустинге, т.к., в предыдущей работе авторов она показала хорошие результаты на сопоставимых объемах данных [8].

Градиентный бустинг — это подход к решению задач путем составления итоговой модели из более простых моделей, например, моделей на основе деревьев решений. При этом каждая новая модель обучается на основе данных об использовании моделей предыдущей итерации, что позволяет снизить погрешность [9]. Алгоритмы на основе деревьев решений являются одними из самых

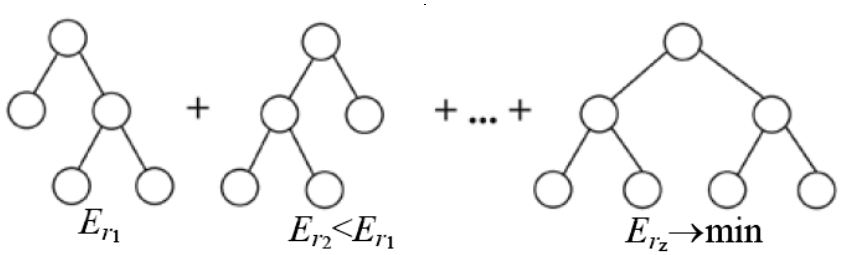

Рис. 1.Минимизация погрешности при использовании алгоритма CatBoost.

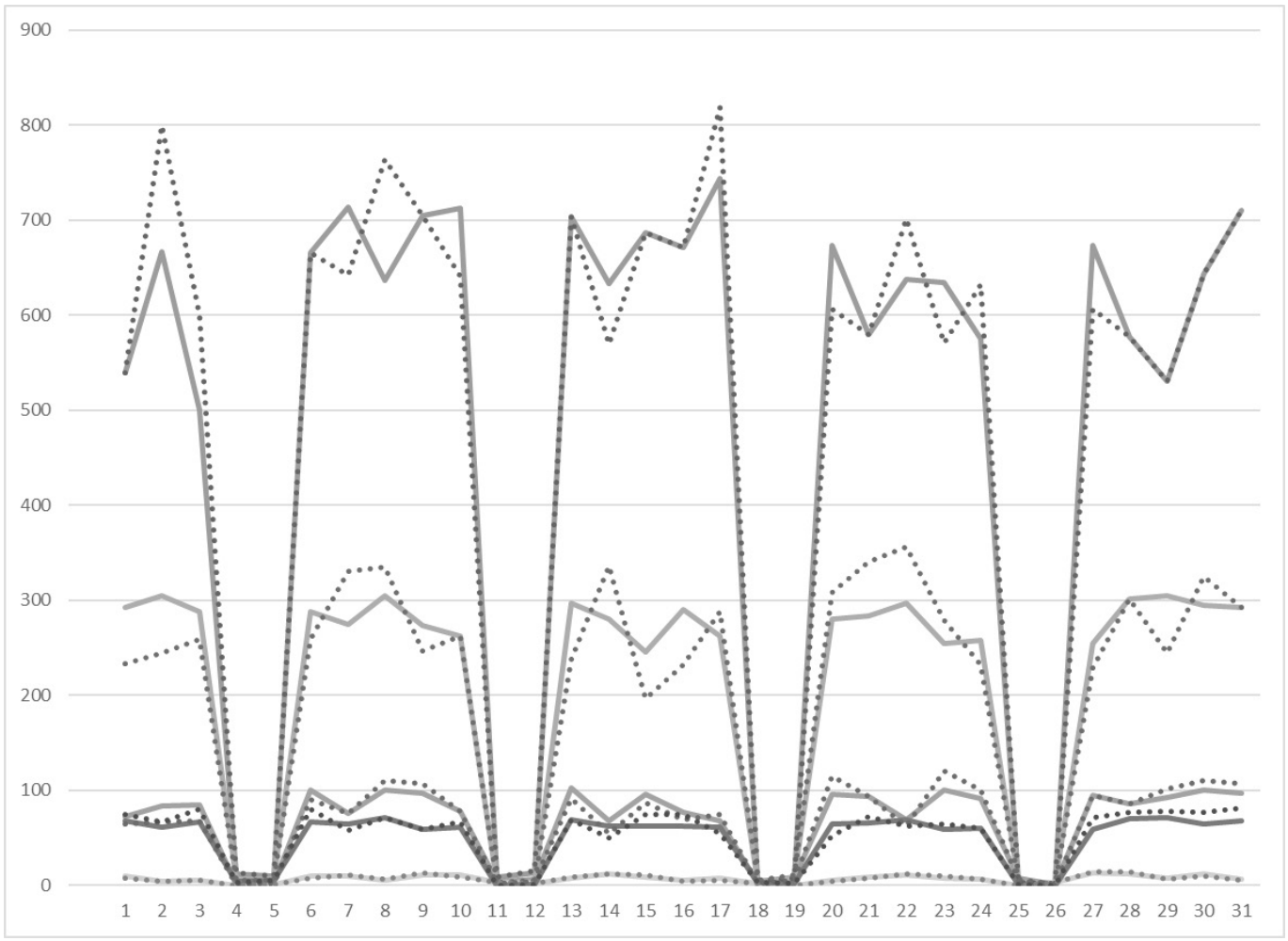

Рис. 2.Сравнение результатов работы модели с реальными данными.

популярных в машинном обучении благодаря их относительно хорошей точности, надежности и простоте использования. Для использования в данной работе был выбран алгоритм CatBoost, представленный компанией Yandex в 2017 году. Особенностью алгоритма является построение симметричных деревьев, возможность работы с категориальными признаками, кроме того, он позволяет обучаться на относительно небольшом количестве неоднородных данных и позволяет минимизировать погрешность путем сложения деревьев (рис. 1). [9]

Эксперимент

Для обучения модели были использованы параметры, установленные в CatBoost по умолчанию, за исключением «Learning rate», «Depth», «N_estimators» и «Количество признаков» = N, где N — общее число параметров для Project. Использовались данные 31 дня разработки, на них же и производилась проверка. После обучения и проверки результаты записывались, после чего один из признаков исключался и обучение повторялось. Ре-

зультаты работы модели графически отображены на рисунке 2: исходные данные указаны сплошными линиями, предсказанные — прерывистыми.

В качестве метода оценки точности предлагаемой модели прогнозирования была использована F1-мера. [11]

$$F = \frac{\frac{2 * TP}{TP + FP} * TP}{\frac{TP}{TP + FN} * TP}$$

$$P = \frac{TP}{TP + FP}$$

$$R = \frac{TP}{TP+FN},$$

где TP — истинноположительное решение,
TN — истинноотрицательное решение,
FP — ложноположительное решение,
FN — ложноотрицательное решение.

Результат прогноза оценивается F1-мерой с оценкой 81% с точностью P=80% и полнотой R=84% (значения округлены до целых чисел). С учетом мнения о том, что прогноз целесообразно строить на основе более чем одной модели, точность этого прогноза считается авторами удовлетворительной, а модель — пригодной для дальнейшего решения задачи после доработки.

Заключение

В данной работе был представлен подход к прогнозированию загрузки вычислительных систем, выражающийся в количестве поступающих за отрезок времени заявок на выполнение различного рода задач.

Прогнозирование нагрузки и затрат важно при планировании и реализации проектов, как связанных с ИТ, так и нет. Получение достаточно точных (авторы считают точность более 80% достаточной) прогнозов помогает принимать управленческие решения как в рамках единичного проекта, так и в рамках управления предприятием, при планировании его развития или выявлении узких мест функционирования. В дальнейшем авторы планируют подробнее изучить вопрос повышения точности модели и затронуть тему интерпретируемых моделей машинного обучения, т.к., по мнению авторов некая идеальная модель прогнозирования должна не только предоставлять результат заданной точности, но и предоставлять объяснение принятых решений или выданного результата.

ЛИТЕРАТУРА

1. Калистратов, А.П., Афанасьев, Г. И. Постановка задачи выбора оптимальной виртуальной машины для решения вычислительных задач предприятия. «Приборостроение в XXI веке — 2019. Интеграция науки, образования и производства» [Электронный ресурс]: сб. материалов XV Всерос. науч.-техн. конф. (Ижевск, 20–22 нояб. 2019 г.).— Ижевск: Изд-во ИжГТУ имени М. Т. Калашникова, 2019.— 345 с
2. Голосовский, М. С. (2015). Информационно-логическая модель процесса разработки программного обеспечения. Программные системы и вычислительные методы, (1), 59–68
3. Калистратов А. П. Методика поиска точки насыщения для однопоточной вычислительной системы. Математические методы в технике и технологиях — ММТТ. 2019. Т. 2. С. 68–73.
4. Ваганова, Е. В., Земцов, А. А., Миньков, С. Л. (2016). Оценка стоимости разработки программного продукта: обзор. Проблемы учета и финансов, (1 (21)).
5. Козадаев А. С., Арзамасцев А. А. Прогнозирование временных рядов с помощью аппарата искусственных нейронных сетей. Краткосрочный прогноз температуры воздуха // Вестник российских университетов. Математика. 2006. № 3.
6. Тушавин В. А. Инженерная методика количественной оценки удовлетворенности потребителей // Информационно-управляющие системы. 2011. № 5.
7. Нетес В. А. Что нужно для успешного применения SLA // T-Comm. 2015. № 7.
8. MACHINE LEARNING IN IT SERVICE MANAGEMENT. Zuev D., Kalistratov A., Zuev A. В сборнике: Procedia Computer Science 9. Сер. "Postproceedings of the 9th Annual International Conference on Biologically Inspired Cognitive Architectures, BICA 2018" 2018. С. 675–679.
9. Ye, J., Chow, J. H., Chen, J., & Zheng, Z. (2009, November). Stochastic gradient boosted distributed decision trees. In Proceedings of the 18th ACM conference on Information and knowledge management (pp. 2061–2064). ACM.
10. Salakhutdinova K. I., Lebedev I. S., Krivtsov I. E. The algorithm for gradient boosting of decision trees in problem of software identification // Scientific and technical journal of information technologies, mechanics and optics. 2018. No. 6.
11. Луценко Е. В. Нечеткое мультиклассовое обобщение классической F-меры достоверности моделей Ван Ризбергена в АСК-анализе и системе «Эйдос» // Научный журнал КубГАУ — Scientific Journal of KubSAU. 2016. № 123.

© Нестеров Юрий Григорьевич (ugn@bmstu.ru),
Калистратов Алексей Павлович (akalistratov@gmail.com), Афанасьев Геннадий Иванович (gaipcs@bmstu.ru).
Журнал «Современная наука: актуальные проблемы теории и практики»