Кафедра математической статистики ИВМиИТ КФУ 6 – 12 января 2020 г.

Научно-популярная статья подготовил Казанцев А.В.

Источник: «Популярная механика» №10, 2017

https://old.elementy.ru/nauchno-populyarnaya_biblioteka/433842/Novyy_um_korolya

«Королева бустинга»[1]

Как создаются лучшие системы машинного обучения в мире

Александр Ершов

У каждой технологической компании есть хорошо известные пользовательские продукты и внутренние разработки, на которых эти продукты держатся. Это своеобразные двигатели, которые вращают шестеренки механизма. Долгое время главным двигателем «Яндекса» была система машинного обучения «Матрикснет», которая обеспечивала и работу поиска, и подбор подходящих рекламных объявлений, и выбор оптимального маршрута в навигаторе. Недавно «Яндекс» завершил работу над новой системой – *CatBoost*, которая призвана полностью заменить «Матрикснет» и стать новым «умом» главного российского поисковика. Разработкой этой системы руководила *Анна Вероника Дорогуш*, недавняя выпускница МГУ, которую с полным правом можно назвать демиургом разворачивающейся на наших глазах компьютерной космогонии.

«Просто я очень люблю решать математические задачки. Ты сидишь над ней час, другой, и когда вдруг начинает складываться, когда части паззла совпадают друг с другом, возникает удивительное ощущение, эйфория. Собственно говоря, с этого все и началось».

[1] Исходное название статьи – «Новый ум короля» – было заимствовано у бестселлера Роджера Пенроуза.

Сейчас Анна Вероника — тимлид одного из самых важных проектов российского поисковика. Но несколько лет назад она была обычной выпускницей, которая зашла на лекцию известного математика, академика Альберта Николаевича Ширяева. Лекцию тогда почему-то отменили, и вместо нее решено было провести семинар для студентов яндексовской Школы анализа данных. «Было очень интересно, а одна из задач оказалась слишком сложной, и ее оставили студентам как домашнее задание. Она меня так зацепила, что очень хотелось ее доделать и показать решение преподавателю, Евгению Бурнаеву. Я не была студенткой Школы и могла только лично попросить его проверить мое решение вместе с другими работами. Но потом ведь надо было вернуться за результатом на следующий семинар, потом еще раз и еще, и так я неожиданно попала в ШАД».

Школа отпраздновала в нынешнем году свое десятилетие. Начиналась она как экспериментальный проект, задачей которого было научить потенциальных соискателей анализировать данные на индустриальном уровне, чего вчерашние студенты обычно не умеют. Сегодня ШАД — это фактически полноценный университет, который бесплатно дает фундаментальное образование. В области машинного обучения и анализа данных Школа может конкурировать с лучшими мировыми университетами, при этом от выпускника не требуют после окончания учебы работать в компании. Некоторые выпускники идут работать к конкурентам, и это считается вполне нормальным.

История Анны Вероники показывает, что часто так и бывает. Учеба в ШАД не помешала ей поработать и в российской компании ABBYY, и в американской *Microsoft*. «Тогда считалось, что надо обязательно уезжать в западную компанию, и это действительно многое мне дало. Но я, как оказалось, очень люблю Москву, поэтому, как и многие мои коллеги, все равно вернулась». Так Анна Вероника оказалась сначала сотрудницей российского *Google*, а потом начала работать в «Яндексе».

Загадка кошкиного зуба

В том, что лучшие специалисты по математическому обучению часто приходят именно в поисковые компании, нет ничего необычного. Ведь поиск — это прежде всего точное соответствие между желанием пользователя и ответом машины. И чтобы научить машины правильно понимать эти желания, нужны специалисты по машинному обучению.

Если отбросить технологический жаргон, то машинное обучение — это просто автоматическая система угадывания. Неважно чего: будущей погоды, котировок акций или адреса веб-страницы. Причем такая система основана не на программировании (когда есть четкий алгоритм поведения), а на демонстрации компьютеру большого числа обучающих примеров. В мире, где информации все больше, машинное обучение часто единственный способ как-то ее осмыслить.

Отличие машинного обучения от программирования очень просто проиллюстрировать: возьмите фотографию кошки и собаки и попробуйте объяснить, как именно вы узнали, кто из них где изображен. Наверняка вы этого сделать не сможете, так как знание о том, что есть кошка, а что собака, вы получили не по формальным правилам, а на опыте. Основано оно на множестве мелких отличий, которые очень сложно выразить словами. Точно так же видит мир и машина, если ее не программировали, а обучали. А вот если бы наше представление о кошках опиралось на парадигму программных кодов, мы могли бы легко ответить, что по формальным признакам кошка от собаки отличается отсутствием

на верхней челюсти второго коренного зуба. Впрочем, навряд ли это помогло бы нам узнать животное по фотографии.

Для крупных IT-компаний, оперирующих петабайтами информации, математическое обучение — основной рабочий инструмент. От него зависит не только работа всех пользовательских продуктов, но и внутренняя кухня: прогнозирование нагрузки на серверы, распределение дискового пространства и т. п. В «Яндексе» до недавнего времени за все это отвечала единая система машинного обучения, введенная в строй еще в 2009 году. Кое-где она дополнялась нейросетями и другими инструментами, но в том или ином виде «Матрикснет» присутствовала во всех продуктах компании.

Идея такой унификации заключалась в том, чтобы внутренние усовершенствования интеллекта «Яндекса» конвертировались в небольшие, но постоянные улучшения функционирования всех остальных сервисов. И до какого-то момента это действительно работало, система оказалась удивительно гибкой. Однако даже многократно оптимизированный и отполированный «Матрикснет» не мог справиться со всеми возложенными на него задачами и избавиться от тех недостатков, что были присущи ему с рождения.

В тени решающих деревьев

«Матрикснет» основан на работе деревьев принятия решений, одного из самых мощных инструментов в мире машинного обучения. Деревья решений — это что-то вроде тех блок-схем, по которым можно определить, «какой ты супергерой». Только рисует их не человек: компьютер перебирает разные варианты организации признаков таким образом, чтобы минимизировать ошибку в примерах с известными ответами. Если полученное дерево построено правильно, то впоследствии оно будет работать и с новыми данными, которые в выборке не встречались.

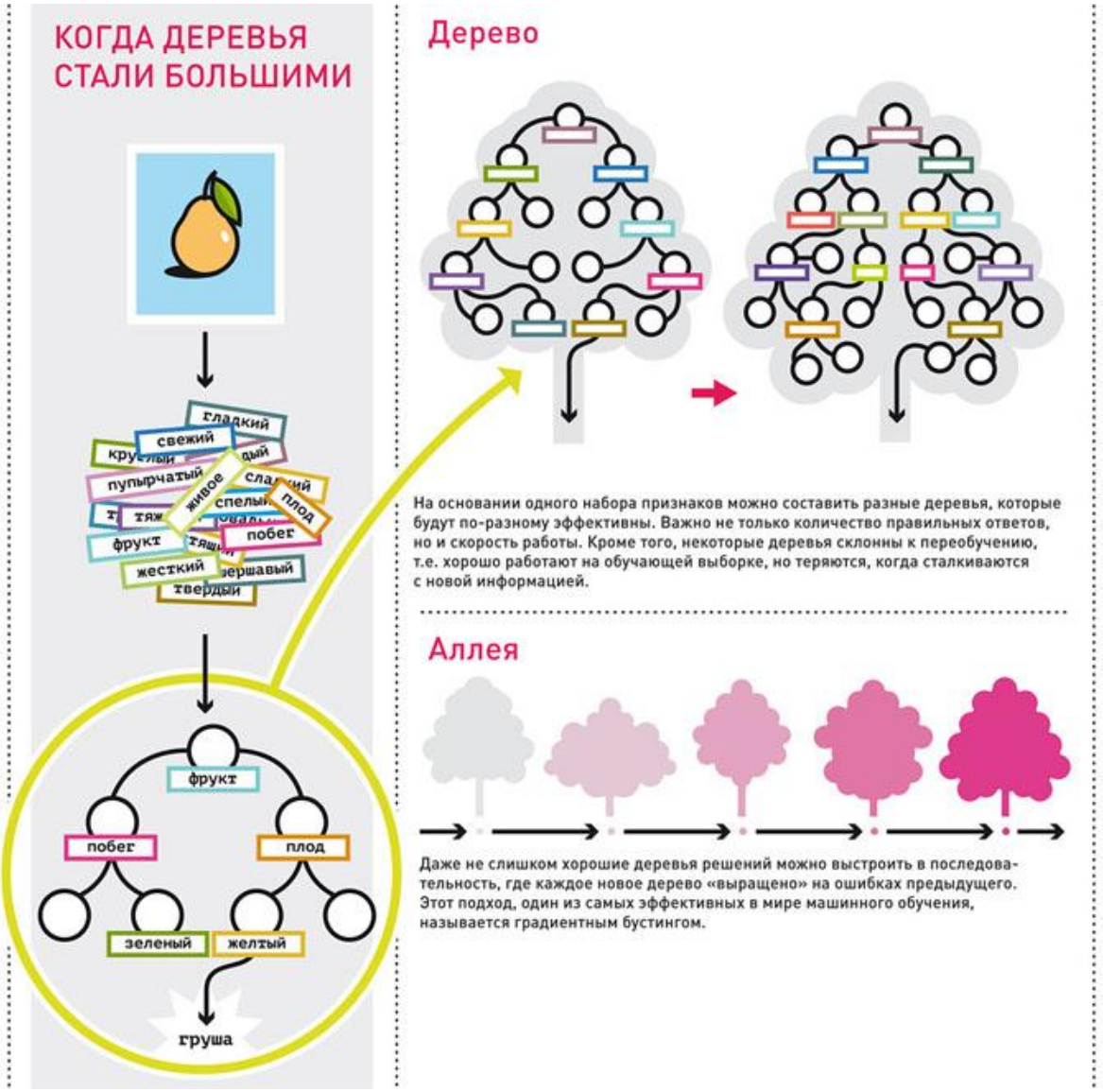
Деревья решений используются для разных задач машинного обучения, но самый очевидный пример — задача классификации. Каждый объект компьютер принимает как набор отдельных, разнородных признаков, которые могу быть как количественными, так и качественными. На основании этих признаков во время обучения строится дерево решений, состоящее из развилок-вопросов и листьев-ответов. Готовое дерево затем используется для того, чтобы машина могла быстро угадывать новые объекты.

Есть, конечно, и другие подходы к обучению — например, всем известные нейросети. Но они хороши прежде всего там, где данные однородны: картинка, звук, видео, текст. Если же нужно построить систему, работающую с произвольными данными, то лучший результат, скорее всего, покажут именно деревья решений.

«Когда я пришла в „Яндекс“, „Матрикснетом“ никто особенно не занимался: считалось, что улучшать там почти нечего. Но на самом деле все оказалось не совсем так, а гораздо интереснее», — вспоминает Дорогуш. Выяснилось, например, что обучение алгоритма можно ускорить в десятки раз. Но еще важнее, что «Матрикснет» не умел грамотно работать с категориальными признаками. Одно дело, когда требуется предсказание на основании чисел — это не всегда простая, но по крайней мере естественная для компьютера задача. Другое дело, когда обрабатываются такие признаки, как вид облаков и тип элементарной частицы (или, например, адрес сайта — это вообще-то тоже

категориальный признак). Таких данных очень много, поэтому хорошая система должна уметь с ними справляться.

Здесь есть несколько стратегий. Можно, например, разделить дерево на столько веток, сколько вообще существует вариантов признака. Или сопоставить каждой категории некое порядковое число, и уже его рассматривать как числовой признак (впрочем, почти бессмысленный). Или ввести новые признаки, количественно описывающие степень принадлежности к той или иной категории. Подходов много, но все они далеки от идеала. Нужно было понять, как научить решающие деревья по-настоящему понимать категориальные признаки и делать это быстро.



Время принятия решения

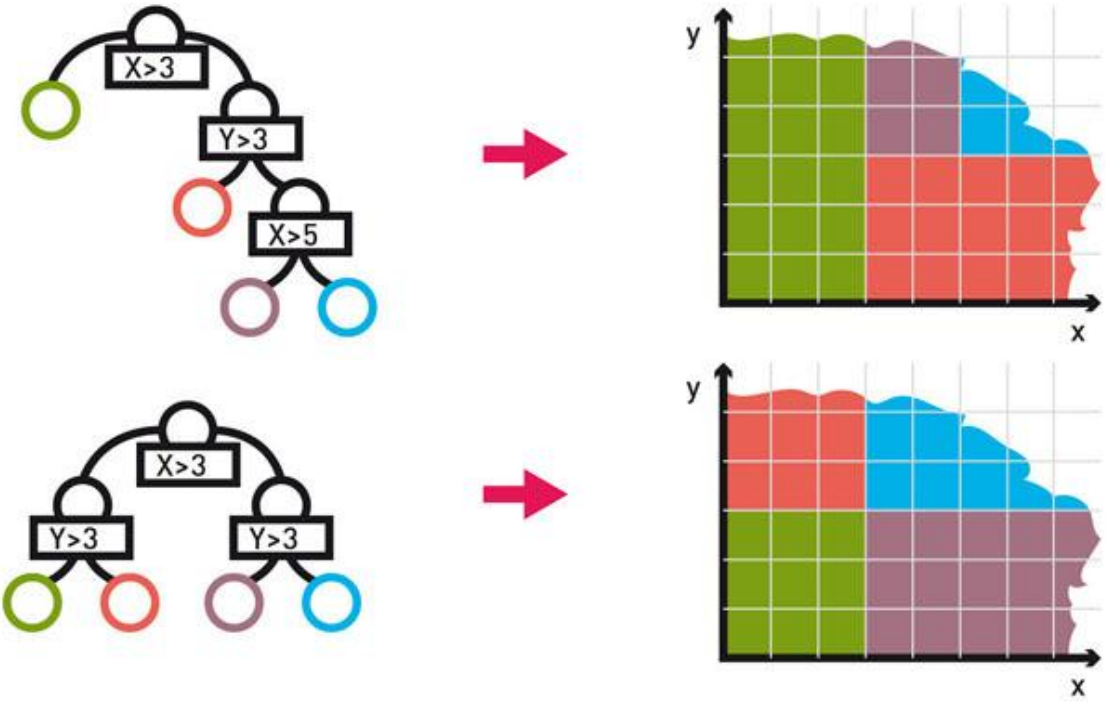
Как раз этой задачей — поддержкой категориальных признаков — в течение нескольких лет занималась команда Андрея Гулина, создателя «Матрикснета». Результатом работы команды была новая версия градиентного бустинга, которая, хоть и была

экспериментальной и не предназначалась для широкого применения, давала результаты лучше, чем сам «Матрикснет».

В основе идеи Гулина лежала новаторская концепция времени, которая позволяла победить главный бич категориальных признаков — склонность к переобучению. Переобучившись, машина ведет себя как школьный зубрила, то есть запоминает наизусть весь учебник (обучающую выборку), но не может ответить на простой новый вопрос. Концепция времени позволяла организовать обучение компьютера таким образом, чтобы в процессе он не мог заглянуть в ответы до конца контрольной, что резко снижало «зазубривание» признаков и стимулировало их понимание.

На базе этой основной идеи, а также других наработок Гулина Анна Вероника и ее команда начали строить новый проект, который мог бы заменить собой «Матрикснет». Его назвали *CatBoost* — от слова «категория» (увы, коты здесь ни при чем).

Почему дерево симметричное?



Существует особый класс деревьев — симметричные, у которых вопросы на каждом уровне ветвления повторяются. Такое дерево может быть легко представлено в виде обычной таблицы с числами — матрицы, что очень важно для скорости вычислений. Компьютеры очень любят работать с матрицами, поэтому справляются с такими деревьями гораздо лучше.

Результаты этой работы, занявшей почти полтора года, можно суммировать простой табличкой. В ней *CatBoost* оставляет позади все доступные на сегодняшний день системы на всех тестовых выборках. При этом алгоритм, в отличие от многих из них, не требует ни ручной настройки, ни какой-либо оптимизации. Первые реальные применения алгоритма показали, например, что он в равной степени подходит и для прогнозирования качества стали, и для определения типа элементарных частиц в CERN, и для поминутного прогноза погоды. Видимо, под впечатлением от таких результатов компания приняла

неожиданное решение выпустить алгоритм под свободной лицензией — теперь строить собственные программы и сервисы на *CatBoost* смогут все желающие.

Когда я спрашиваю Анну Веронику о том, приятно ли ей чувствовать себя создателем чего-то самого-самого, она говорит, что, конечно, очень гордится результатом, но не только этим: «Я горжусь еще и тем, что нам помогали ребята из самых разных команд „Яндекса“ — просто так, не по работе, а из интереса. Все за нас болели, все хотели, чтобы мы смогли представить миру лучшую в своем классе систему машинного обучения. И у нас это получилось».