



> Конспект > 0 Урок > Прикладная разработка на Python

> Оглавление

- > Оглавление
- > Что такое Machine Learning
- > Примеры применения ML
- > Специфика ML
- > Выбор инструмента для работы с ML
 - Офисные пакеты (MS Office, LibreOffice и т.д)
 - Языки программирования
- > Почему Python?

> Что такое Machine Learning

В настоящее время существует множество определений Machine Learning, которые так или иначе похожи.

Одним из первых, кто использовал термин «Machine Learning», был изобретатель первой самообучающейся компьютерной программы игры в шашки А. Л. Самуэль в 1959 г:

"Machine Learning — это процесс, дающий компьютерам способность обучаться новому, не будучи непосредственно запрограммированными делать это."

(A.L. Samuel Some Studies in Machine Learning Using the Game of Checkers // IBM Journal. July 1959. P. 210–229.)

Более точное определение дал позднее Т.М.Митчелл:

"Программа обучается с опытом E решению некоторой задачи T по метрике качества P , если качество ее решения T , измеренное согласно P , растет вместе с ростом опыта E ."

(*T.M. Mitchell Machine Learning. McGraw-Hill, 1997.*)

> Примеры применения ML

- **Распознавание образов**

Выделение конкретных объектов на изображении или видео (люди, машины, мебель и т.д).

- **Определение интересов пользователя**

Google и Яндекс также применяют в своей работе машинное обучение, чтобы показывать рекламу определенным пользователям. Если вы хоть раз замечали, что после поиска информации о каком-то товаре почти сразу же видели соответствующее объявление в поисковиках, то это было сделано благодаря технологиям машинного обучения.

- **Предсказание курса валют**

На основе статистических данных можно попытаться уловить закономерность и примерно предсказать, куда будет двигаться валюта. Это актуальная задача для финансовых институтов.

- **Распознавание слов в аудиозаписи**

Субтитры – это отличный способ, чтобы увеличить охват аудитории. YouTube использует технологию распознавания речи, чтобы автоматически создавать субтитры к видео. Субтитры автоматически генерируются алгоритмами самообучающейся системы, так что их качество может отличаться от фильма к фильму.

- **Определение мошенничества**

Также актуальная задача во всех крупных банках: банкам требуется определять, является ли транзакция мошеннической. Например, совершена ли покупка незаконным способом или перевод денежных средств не соответствует тому, что указано в его описании. Так как в день проходит большое количество транзакций, банкам необходимо определять это каким-либо автоматическим способом.

> Специфика ML

- **Появление новых подходов в короткие сроки**

Статьи, исследующие новые методы машинного обучения, публикуются довольно часто. Количество новых подходов постоянно увеличивается.

- **Масштабируемость**

Алгоритмы должны адаптироваться под возрастающую нагрузку и продолжать работать с тем же или лучшим качеством.

- **Сочетаемость методов**

Желательно использование не только базовых методов, но и их комбинация с более продвинутыми или новыми методами так, чтобы это шло не в ущерб финальному качеству.

> Выбор инструмента для работы с ML

Офисные пакеты (MS Office, LibreOffice и т.д)

- *Устаевают в день релиза* — при появлении новых алгоритмов машинного обучения офисные пакеты не будут обновлены вслед за их выходом. В лучшем случае, это произойдет через год или два, а в худшем — поддержка новых алгоритмов совсем не будет добавлена.
- *Плохо масштабируются* — при значительном увеличении трафика обработка данных с использованием офисных пакетов становится проблематичной.
- *Изначально проектировались для других целей* — целью офисных пакетов, как правило, является составление отчетов и анализ небольшой порции данных.

Языки программирования

- Можно быстро реализовать новый алгоритм. Из-за открытости языков программирования новые алгоритмы могут быть распространены среди разработчиков в короткие сроки.
- В большинстве языков сразу закладывается возможность увеличения данных со временем.
- Оставляют свободу выбора в их использовании.

> Почему Python?

- **Универсальность**

Изначально не создавался под определенную задачу. Изначально позиционировался как "язык для каждого".

- **Простота языка**

Понятный синтаксис и низкий порог вхождения

- **Большая скорость разработки**

Время, затрачиваемое на реализацию той или иной идеи, меньше из-за того, что разработчик фокусируется на самом алгоритме, а не на особенностях языка

- **Развитая экосистема**

Существует множество библиотек под разные нужды

- **Гибкость в изменениях**

В Python можно изменить поведение практически любой конструкции