

Искусственный интеллект: Глубокое обучение

7 причин почему MATLAB самая простая и продуктивная среда для инженеров и учёных

1. Матлаб говорит на языке математики

Инженеры и ученые нуждаются в языке программирования, который позволит сконцентрироваться на исследованиях, а не на изучении языка программирования. Линейная алгебра в MATLAB выглядит как линейная алгебра в учебнике. То же самое относится и к анализу данных, обработке сигналов и изображений, разработке элементов управления и т.д.

2. Наборы инструментов MATLAB просто работают

Инструментарий MATLAB предлагает профессионально разработанные, тщательно протестированные и полностью документированные функциональные возможности для широкого применения для научных и инженерных задач. Инструменты MATLAB предназначены для совместной работы и интегрируются с параллельными вычислительными средами, графическими процессорами и генерацией кода на языке C.

3. MATLAB разработан специально для инженеров и ученых

Названия функций и подписи знакомы и запоминаемы. Среда рабочего стола настроена для итеративного проектирования и научных исследований. Документация написана для инженеров и ученых, а не для программистов.

4. MATLAB интегрирует рабочие процессы

Основные инженерные и научные проблемы требуют широкой координации между командами для реализации идей. Каждая передача по пути добавляет ошибок и задержек. MATLAB помогает автоматизировать весь путь от исследований до производства

5. MATLAB это быстро

Ускорение создания вашего кода посредством распределения математических операций по ядрам вашего компьютера, библиотечные вызовы оптимизированы, и весь код компилируется точно в срок.

6. MATLAB имеет приложения

Приложения MATLAB - это интерактивные приложения, которые сочетают прямой доступ к большим коллекциям алгоритмов и немедленную визуальную обратную связь. Вы можете сразу увидеть, как разные алгоритмы работают с вашими данными. Повторяйте, пока не получите желаемые результаты, затем автоматически сгенерируйте программу MATLAB для воспроизведения или автоматизации вашей работы.

7. MATLAB доверяют

Инженеры и ученые по всему миру доверяют MATLAB используя его в процессе запуска космических аппаратов, подбирая пациентов с трансплантацией и доноров органов или просто составляя отчет для управления. Это доверие основано на безупречных числах, вытекающих из работы MATLAB в сообществе исследователей численного анализа.

Команда инженеров MathWorks постоянно проверяет качество, выполняя миллионы тестов на базе кода MATLAB каждый день.

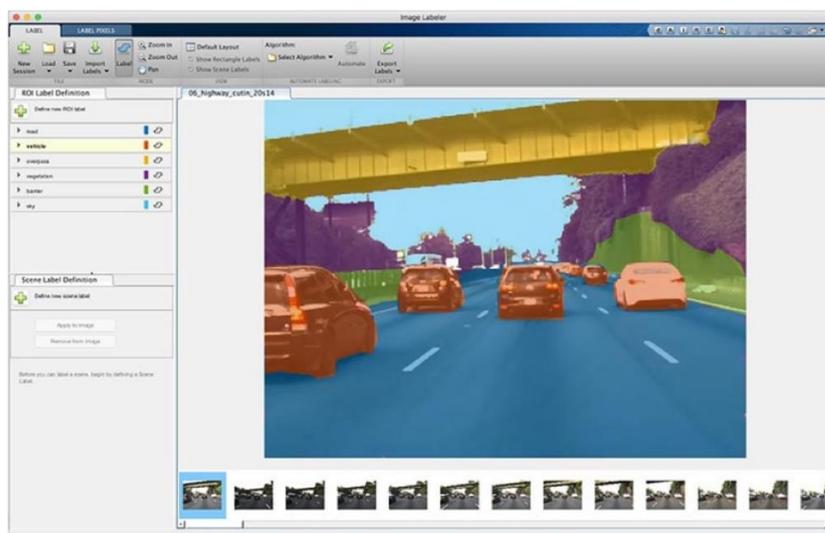
Почему MATLAB подходит для глубокого обучения

С помощью всего лишь нескольких строк кода MATLAB или на основе интерактивного инструмента вы можете создавать модели глубокого обучения.

- Легко получить доступ к самым последним моделям, включая GoogleNet, VGG-16, VGG-19, AlexNet, ResNet, Inception, InceptionResNet и DenseNet.
- Ускорение алгоритмов на графических процессорах от Nvidia®, использование облаков и ресурсов центров обработки данных без специализированных программ.
- Создание, модификация и анализ сложных архитектур глубоких нейронных сетей с помощью MATLAB apps и инструментов визуализации.
- Автоматизация маркировки изображений, видео и аудио с помощью интерактивных приложений.
- Работа с моделями Caffe и TensorFlow-Keras.
- MATLAB поддерживает ONNX, поэтому вы можете сотрудничать с коллегами, использующих такие фреймворки, как PyTorch и MxNet.

Совместимость

Это не выбор между платформами MATLAB и Python. MATLAB поддерживает взаимодействие с платформами глубокого обучения с открытым исходным кодом, использующими возможности импорта и экспорта ONNX. Используйте инструменты MATLAB там, где это наиболее важно – доступ к возможностям и предварительно подготовленным функциям и приложениям, недоступным в Python.



Мультиплатформенное портирование алгоритмов

Развертывайте модели глубокого обучения на различные платформы включая CUDA, C-код, корпоративные системы или облака. Когда производительность имеет значение, вы можете генерировать код, который использует оптимизированные библиотеки от Intel® (MKL-DNN), NVIDIA (TensorRT, cuDNN) и ARM® (APM вычислительные библиотека) для создания модели глубокой сети с высокой скоростью работы.



Основные этапы рабочего процесса в системах, основанных на глубоких сетях состоят из:

1. Создание базы данных для обучения
2. Разработка архитектуры
3. Оптимизация параметров сети
4. Оптимизация гиперпараметров и валидация решения
5. Перенос алгоритма на целевую платформу

Глубокое обучение в задачах компьютерного зрения

MATLAB — это среда в которой вы легко можете собрать и протестировать архитектуру под конкретную задачу компьютерного зрения, такую как распознавание образов или детектирование объектов на изображении.

Распознавание образов

Для распознавания образов вы можете использовать уже [обученные сети](#) с помощью технологии [передачи обучения](#), или создавая свои собственные архитектуры как в виде программного кода, так и с помощью интерактивного инструмента [Deep Networks Designer](#)

[Пример использования техники передачи обучения](#)

Детектирование объектов

Детектирование объектов — это технология в основе которой лежит техника распознавания образов. Выделяют два подхода детектирования:

1. На основе [выделения регионов](#) в которых находится предполагаемый объект
2. Попиксельная сегментация: [пример](#) и видео во вложении: segnet1.mp4

Глубокое обучение: естественные языки

MATLAB это удобная среда для работы с текстом.

На основе глубоких сетей вы можете [классифицировать](#) и [генерировать](#) текст.

Глубокое обучение: обработка аудио сигнала

- [Распознавание речи](#)
- [Удаление шума](#)

Глубокое обучение: временные ряды

- [Предсказание поведения временного ряда](#)
- [Классификация последовательности Sequence-to-Sequence](#)

Глубокое обучение: обучение с подкреплением

Обучение с подкреплением — это техника в которой алгоритм учится по принципу проб и ошибок. Возможности для обучения с подкреплением появились в MATLAB в релизе R2019a (март 2019 года) и содержат такие алгоритмы как Deep Q-Network, Deep Deterministic Policy Gradients, Advantage Actor Critic и другие.

Примеры:

- [Создание агента с помощью Deep Network Designer и обучение с использованием наблюдений изображений](#)
- [Обучение с подкреплением на сетке](#)
- [Обучение с подкреплением для перевернутого маятника](#)

Глубокое обучение: диагностическое обслуживание

Диагностическое обслуживание становится все более актуальной тематикой. Это связано с появлением датчиков на оборудовании и большими наборами информации с этих датчиков. Один из успешных методов, применяемых для предсказаний поломок является [глубокое обучение](#).

Создание базы данных для обучения

Для задач глубокого обучения наличие качественной базы данных является ключевым атрибутом успешного решения задачи. Образно говоря, качественная и представительная база данных - это 90% успеха. База данных включает в себя 3 компонента: сырые данные, разметка и объект для доступа к данным. В этот этап, в некоторых случаях, включают предобработку данных: выделение характеристик, предобработку текста и пр. Хотя выделение характеристик относится к задачам машинного обучения, в задачах глубокого обучения их применяют для обработки последовательностей, таких как показания датчиков, аудио сигналов и т.д.

Сырые данные - это данные собранные в результате испытаний (показания датчиков, видеоряд с веб камеры и т. д.). В общем случае, чем больше сырых данных, тем лучше.

Разметка данных – процесс, ставящий в соответствие объект сырых данных с некоторым классом (например, изображение кота ставится в соответствие классу «кот» или «млекопитающее»). Разметка, как правило, производится вручную, однако, в MATLAB можно использовать интерактивные инструменты, которые позволяют автоматизировать процесс разметки.

Доступ к базе данных - это специальный класс, который позволяет оптимальным способом обращаться к базе данных при обучении нейронной сети. В зависимости от типа базы данных в MATLAB реализованы следующие объекты:

- [datastore](#) — объект для работы с данными общего вида

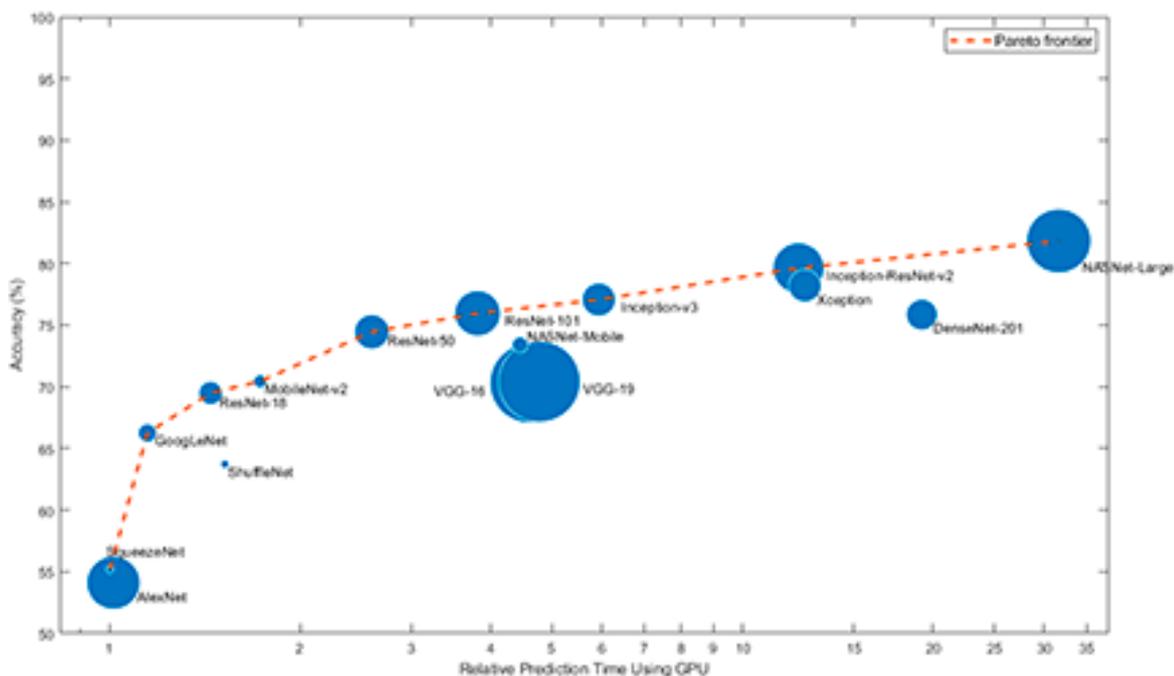
- [imageDatastore](#) — объект для работы с изображениями
- [audioDatastore](#) — объект для работы с аудио файлами

[Автоматизированная маркировка данных для обучения классификаторов в MATLAB](#)

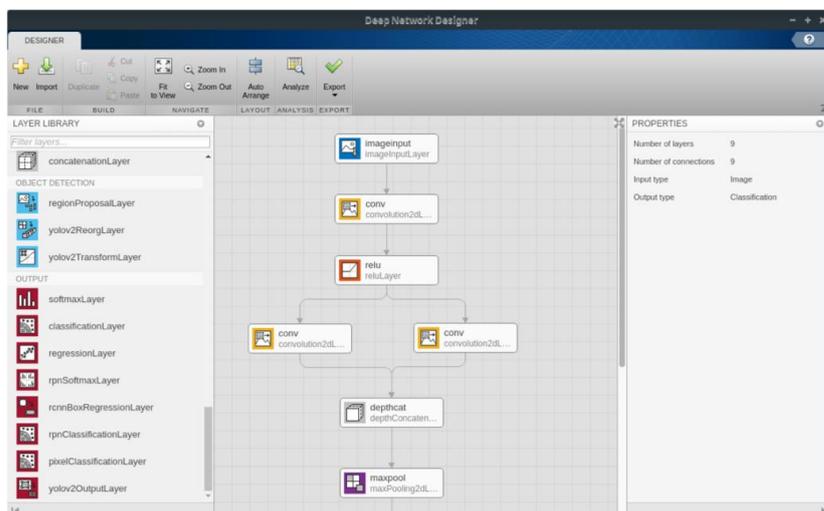
[Автоматизация при создании базы изображений](#)

Разработка архитектуры

Разработка архитектуры - 2-й по важности компонент после базы данных. В большинстве случаев используют уже готовую архитектуру из множества обученных нейронных сетей, дообучая их под свою задачу. Этот процесс называется передачей обучения. Он в основном применим для задач компьютерного зрения. Список доступных сетей вы можете посмотреть и скачать [здесь](#).



Процесс передачи обучения описан в [документации](#). Его удобно проводить с помощью интерактивного инструмента Deep Network Designer.



Этот инструмент также полезен для создания своих архитектур, т.к. обладает свойством наглядности, анализа, выявления ошибок, а также вы можете генерировать m код для дальнейшего использования в процессе обучения.

[Глубокое обучение в MATLAB](#)

[Deep Learning: 1. Слои глубоких сверточных сетей](#)

[Глубокое обучение с MATLAB - Подготовка данных](#)

Оптимизация параметров сети (обучение сети)

Оптимизация параметров сети является ресурсозатратным этапом. В подавляющем большинстве случаев сети обучаются на видеокартах NVIDIA, используя при этом распределенные вычисления в облаках или на собственном кластере. Однако, предварительное обучение и выбор параметров обучения, таких как скорость обучения, тип алгоритма, размер выборки на каждом шаге и т. д. удобно проводить на десктопной машине. Для отслеживания процесса обучения в MATLAB используется инструмент [мониторинга](#). Он предназначен для визуализации и преждевременной остановки процесса обучения, что позволяет эффективно настраивать параметры обучения.

[Глубокое обучение с MATLAB - Оптимизация параметров и гиперпараметров сети](#)

Оптимизация гиперпараметров и валидация решения

К гиперпараметрам сети относятся компоненты, которые не настраиваются в процессе обучения. На пример, это может быть количество слоев в сети, количество нейронов в конкретном слое, скорость обучения и т. д. От правильного подбора гиперпараметров зависит качество обучения и, соответственно, точность распознавания (классификации или регрессии).

Гиперпараметры можно настраивать перебором, но это очень затратная процедура. Одним из современных способов настроить гиперпараметры — [Байесовская оптимизация](#). Она позволяет добиться приемлемых результатов минимизируя при этом количество запусков обучений сети.

[Глубокое обучение с MATLAB - Оптимизация параметров и гиперпараметров сети](#)

Перенос алгоритмов на целевую платформу

В зависимости от требований, MATLAB предоставляет различные возможности для переноса алгоритмов глубокого обучения на целевые платформы. Это может быть создание независимого от MATLAB приложения, но работающего на основе библиотек MATLAB. Другой вариант, генерация C/C++ кода на микропроцессоры для различных архитектур или CUDA кода для Nvidia GPU и NVIDIA Jetson. На конец, генерация Verilog/VHDL кода для запуска на ПЛИС. Все описанные возможности максимально автоматизированы и занимают минимум времени.

Ключевые возможности:

- Автоматическое преобразование обученных моделей глубокого обучения из MATLAB кода в CUDA с помощью GPU Coder™
- Интеграция [сгенерированного CUDA кода с NVIDIA® TensorRT](#)
- Поддержка DAG (ациклические направленные графы) сетей, [включая GoogleNet, ResNet-50, ResNet-101, и SegNet](#)
- Создание кода на основе обученных моделей глубокого обучения для процессоров Intel® Xeon и ARM® Cortex-A
- Автоматизированное развертывание на платформах NVIDIA Jetson и DRIVE

- Оптимизация глубокого обучения: улучшенная производительность благодаря автоматической настройке, слиянию слоев и поддержке Thrust библиотеки

Услуги:

- Сопровождение проекта
- Консультации при создании базы данных
- Настройка рабочего процесса
- Выбор архитектуры сети
- Настройка параметров и гиперпараметров сети
- Перенос кода на целевые платформы (получение оптимального C/C++, CUDA, HDL кода из кода MATLAB)

[Генерация C кода из MATLAB](#)

[Глубокое обучение с MATLAB - Генерация GPU кода](#)



ЭКСПОНЕНТА
ЦЕНТР ИНЖЕНЕРНЫХ ТЕХНОЛОГИЙ
И МОДЕЛИРОВАНИЯ

Контакты

exponenta.ru

E-mail: info@exponenta.ru

Тел.: +7 (495) 009 65 85

Адрес: 115088 г. Москва,

2-й Южнопортовый проезд, д. 31, стр. 4