Сеть готова, что дальше?

Лекция

Материал лекции размещен -
https://github.com/d-yacenko/NN_to-_production_impl.git

Лекция предназначена для студентов курса Samsung IT academy, трек ИИ. Она представляет собой с одной стороны практическое занятие по разработке простых прикладных решений основанных на ИНС, а с другой представляет собой обзор различных интеграционных решений, в которых применяется или может применяться ИНС.

Цель

Создание алгоритма использующего ИНС. Цель и задачи.

- прикладная цель — решение конкретной технической проблемы с характеристиками решения не хуже существующих аналогов. В этой ситуации на выходе должно быть прикладной продукт — готовое к использованию в целевом юзкейсе ПО или программно-аппаратный комплекс
- демонстрация — например:
  - концепт демонстрирующий потенциально возможное продуктовое решение — например презентация идеи стартапа,
  - концепт демонстрирующий скилы создателя — портфолио, учебные цели - индивидуальны проект
  - научная работа, лишь демонстрация идей или возможностей, концепт не обязателен. Реализация предмета исследования — научная новизна/учебный продукт. В качестве продукта научная статья или учебные материалы.

Достижение

Вспомним, что работа с НС это learning/inference, соответственно и работа с сетью разделяется на две части. Во всех случаях кроме последнего (рассмотрим чуть позже) требуется реализация двух разных решений — (1) разработка алгоритма ИНС, решающего поставленную задачу со всеми сопутствующими условиями см. первую лекцию , (2) разработка прикладного приложения, соответствующего требованиям к ПО.
(1) Первая стадия состоит из:

- проектирования компонентов нейронной сети: Архитектура НН, Функция потерь, Метод оптимизации, Метрики
- реализации сети и ее обучении: выбор библиотек, создание пайплайна, выполнение процесса обучения на пайплайне, анализ полученных результатов, корректировка сети/рассмотрение новых гипотез [1]
- фиксации результатов в виде документации, в виде опубликованного проекта, экспорт сети для дальнейшего использования

Примечание 1: первая стадия при реализации научной работы должна иметь признак научной новизны, теоретической значимости исследования, и должна удовлетворять критериям научности.
Примечание 2: первая стадия при реализации образовательной работы (методичка, пособие, учебный курс, демонстрационные проекты) в основном служит для демонстраций алгоритмов НС.

(2) Вторая стадия состоит из:

- анализа постановки прикладной задачи и выбор технологической платформы для реализации
- проектирование и разработка ПО использующего сериализованную НС для инференса по фактическим данным.
- имплементация прикладной системы как ПО и/или библиотек/фреймворков и/или сервисов и/или программно аппаратных комплексов

Примечание 1: Вторая стадия для случая научной работы — публикация статьи в научном журнале, участие в конференции.
Примечание 2: Вторая стадия для случая образовательной работы — публикация и внедрение образовательного материала.

Пример

Пример реализации цели «Индивидуальный проект IT Академия Samsung трек ИИ определение пожара на фотографии»
Поскольку эта цель подпадает под (портфолио, учебные цели) задачи:

1. получить датасет, спроектировать, реализовать, обучить НС, анализ и отчет,
2. спроектировать, реализовать, опубликовать концепт.

Реализуем первую стадию.

Проектирование.

Датасет.
Получаем датасет со следующими параметрами:

- две части train(80%)/val(20%)
- два класса фотографий fire(91)/nofire(138)
- состав — фото из разных источников 3872x2592 — 300x214

Вывод:

- датасет слишком маленький для полноценного обучения,
- датасет скошенный,
- требуется предобработка фотографий — размер, обогащение.

Сеть.
Компоненты:

- Т.к. датасет маленький, используем transfer-learning - берем предобученную сеть Resnet18 (обучен на ImagNet ~15M) без fine-tunig
- К выходу сверточных слоев ResNet (признаковое описание ImageNet) подключаем полносвязный слой 512x2
- Некоторый трюк - CrossEntropyLoss
- SGD
- Метрики — Loss и Accuracy

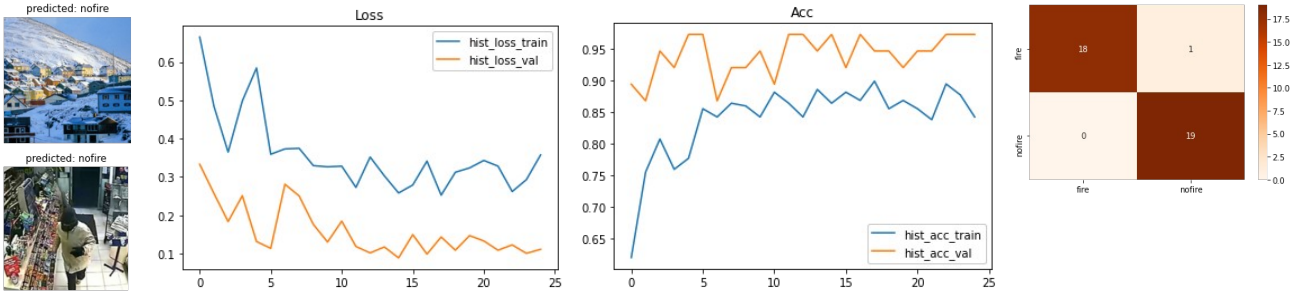
Реализация.

- пайплайна фактически нет, т.к. предполагаем что угадаем сеть с первого раза.
- среда исполнения ноутбук Google Colab,
- фреймворк pytorch
- предобученная модель — torchvision.models.resnet18

Демонстрация. Приложение 1.

Результаты.

Training complete in 2m 0s
Best epoch:14 val Loss:0.088741 Acc: 0.973684
На таком маленьком датасете сеть обучилась очень быстро (обучали то два нейрона), точность 97% - выглядит неплохо, хотя на таком не репрезентативном датасете недостоверна и валидация. По хорошему нужно собрать отдельные данные для оценки достоверности результатов.



Забираем сериализованную модель — файл **Fires.pt**

Реализуем вторую стадию.

Анализ постановки прикладной задачи

- Если задача научная или учебная, на этой стадии оформляют весь собранный на предыдущих этапах материал в виде статьи/метод.материала с учетом требований. В качестве демонстрации можно рассмотреть любую статью в научном журнале [3]
- Если речь о концепте необходимо продумать форму удобной для представления, но с учетом минимальной функциональности и минимальной трудоемкости. Например чат-бот телеграмм. Демонстрация, Приложение 2.
- Если требуется разработка прикладного решения нужно рассмотреть требованию к решению и параметры результата, получившегося на первом этапе :
  - параметры инференса на полученной сети — время на один цикл,
  - среда исполнения,
  - требования по скорости инференса,
  - требования по оперативной памяти,
  - по системе вычислений (TOPS/FLOPS),
  - по программной платформе,
  - архитектура целевой системы,
  - характеристики потока данных для инференса
  - требования безопасности.

По результатам анализа выбирается архитектура прикладного решения, используемые фреймворки и выполняется уже непосредственно проектирование АО и разработка.

Примеры анализа требований и архитектуры решений:
1) Приложение для конечного пользователя, оценка одежды на человеке.

Анализ:

- приложение не требует реального времени работы,
- скорее всего пользователь будет не рад отправки фотографий на сервер,
- поток данных для инференса низкий,
- предположительное место работы — сотовый телефон.

Предлагаемое решение:

- создать мобильное приложение,
- встроить полученную на первом этапе ИНС в приложение для инференса,
- поскольку время на инференс не ограничено, этот параметр скорее всего не критичен,

- если размер сети слишком велик (порядка 100M и выше), то применить квантизацию [2].

Демонстрация. Приложение 3.

2) Приложение для крупной организации, например распознавание fraudulent транзакций или фишинга.

Анализ:

- записей в крупной организации очень много — от 1млн и больше,
- данные обычно хранятся в DWH,
- требование по скорости инференса — ASAP,
- крупные организации обычно не сильно стеснены в объемах памяти и специализированных вычислителях,
- данные для инференса поставляются из БД,
- требования безопасности максимальные — закрытый информационный контур.

Предлагаемое решение:

- то время обработки, то скорость одного инференса критична, т.е. требуются все возможные ухищрения по повышению скорости обработки — дистиляция, квантизация, перевод на специализированные вычислители (миллионы записей!),
- можно получать данные от DWH через какой либо коннектор в сервер инференса в контуре безопасности, а результаты загружать обратно,
- а можно, если хранилище данных построено по архитектуре MPP (Greenplum/ADB) и имеет встроенную среду исполнения программ, выполнять инференс прямо на узлах MPP, что увеличит скорость еще в сотни раз.
- Кстати возможно использовать этот же MPP кластер и для быстрого обучения [4], [5].

3) Распределенная система видеонаблюдения/охранная система.

Анализ:

- большое количество камер наблюдения направлены на территорию и предназначены для уведомления о проникновении,
- уведомление должно быть близким к реалтайму (<1сек)
- требования по безопасности не допускают передачу данных за пределы контура охраны,
- в местах размещения камер нельзя установить стационарное оборудование,
- поскольку камеры не обладают вычислительными возможностями, архитектура системы естественным образом будет клиент-серверной в рамках контура безопасности,
- серьёзных ограничений на вычислительные ресурсы нет.

Предлагаемое решение:

- можно видеопоток от всех камер свести на один сервер и делать там инференс, для обнаружения проникновения нарушителей, правда нагрузка на сеть передачи данных и особенно на сервер будет большая,
- альтернативно, можно воспользоваться архитектурой EDGE и установить небольшие (и недорогие) вычислители например RpI или Nvidia Jetson (квантизация!), на которых вести инференс, а результаты отправлять на сервер, тем самым снимая нагрузку [7].
- можно усовершенствовать решение выше разделив инференс на две стадии — на EDGE выделение карты признаков (например pretrained resnet) и передача на сервер, а на сервере сеть делает вторую фазу инференса. Почитать про двухстадийный инференс можно тут [6].

4) Системы IoT — десятки датчиков, из тысяч местоположений передают информацию в аналитический центр, для построения статотчетов и проактивного анализа.

Анализ:

- Данные от датчиков по протоколу MQTT или по какому либо другому (иногда кастомному) через череду систем — брокеры (возможно EDGE Mosquito), стриминговые сервисы (NiFi/ADS), ELT/ETL загружается в DWH, где над аккамулированными данными производятся аналитические операции — нормализация, агрегация и подготовка операционного слоя данных и витрин данных,
- Витрины данных выводятся заказчикам информации (отделы аналитики, клиентский веб сайт) а в операционном слое работает проактивный анализ.

Предлагаемое решение:

- На операционном слое, как и в примере (2) можно применить ИНС, например как функцию на ЯП встроенном в СУБД,
- возможно в некоторых случаях проводить первую стадию инференса на EDGE, как в примере (3)

Вывод

Как и всегда разработка наукоемких технических решений требует опыт из различных областей деятельности. Ошибкой было бы думать, что требуется только навыки проектирования ИНС. Кроме этого требуются умение проектирования и разработки ПО, навыки системной интеграции, а также опыт внедрения и обслуживания ИС.

Литература

1. Роман Суворов. Как эффективно проводить эксперименты: базовая структура проекта, процесс перебора гипотез, трюки для обучения нейросетей [Электронный ресурс]/Роман Суворов, 2020г - https://www.youtube.com/watch?v=RS_U6qodpsc&t=1s
2. Алексей Ивахненко. Мобильные архитектуры нейросетей и фреймворки для их запуска [Электронный ресурс]/ Алексей Ивахненко, 2020г - https://www.youtube.com/watch?v=ASChrJhj-zY&t=11s
3. TF-IDF vs Word Embeddings for Morbidity Identification in Clinical Notes: An Initial Study/ Danilo Dess, Rim Helaoui, Vivek Kumar1, Diego Reforgiato Recupero and Daniele Riboni University of Cagliari, Cagliari, Italy{danilo dessi, vivek.kumar, diego.reforgiato, riboni}@unica.it Philips Research, Eindhoven, Netherlands rim.helaoui@philips.com
4. Michiel Shortt. AI/ML, Neural Networks & the future of analytics: Using Greenplum and PL/Python [Электронный ресурс]/Michiel Shortt , 2020г. - https://www.youtube.com/watch?v=6ayC8eOamkw&list=PL4duir3J-8GW-P1RXN67IgXMS3LoHXG6u&index=2 , A42 Labs.
5. Bastiaan Sjardin. Large Scale Machine Learning with Python/Bastiaan Sjardin, Luca Massaron, Alberto Boschetti — Packt Publishing,2016
6. CATHERINE SANDOVAL, Two-Stage Deep Learning Approach to the Classification of Fine-Art Paintings / CATHERINE SANDOVAL , ELENA PIROGOVA, MARGARET LECH - School of Engineering, RMIT University, Melbourne, VIC 3000, Australia, 2019
7. Ультимативное сравнение embedded платформ для AI [Электронный ресурс]/ZlodeiBaal, 2019г - https://habr.com/ru/company/recognitor/blog/468421/

Приложение 1.

Алгоритм ИНС.

https://github.com/d-yacenko/NN_to-_production_impl/blob/main/Annex1_fire_fc.ipynb

Приложение 2.

Пример чат-бота.
https://github.com/d-yacenko/NN_to-_production_impl/tree/main/tg_bot

Приложение 3.

Пример андроид приложения.
https://github.com/d-yacenko/NN_to-_production_impl/tree/main/FireRecognize