Генерация данных

Введение

Первым шагом на пути к решению задачи является поиск/генерация данных для дальнейшего обучения, тестов, проверок гипотез.

Чем данные будут более правдоподобные, тем точнее можно будет сказать о том, получается ли решать задачу, какой из подходов работает лучше и прочее. В процессе разработки генератор скорее всего будет дорабатываться и, главеное, расширяться, становясь более правдоподобным.

Нужно сделать так, чтобы на выходе могли получаться реалистичные и разнообразные данные. Разнообразие здесь важно, так как, если будет много типичных данных, то модель будет сильно переобучаться, или находить паттерны, не подходяшие для решения

задачи в общем случае.

Решено было разделить задачу на два этапа: создание генератора искусственных данных и создание функционала для получения (через апи или парсинг) реальных данных из каких-то источников.

Искусственный генератор данных

Для создания искусственного генератора данных использовалсь библиотека Faker. Она содержит в себе много провайдеров, генераторов для создания правдоподобных случайных данных.

Для того, чтобы в полной мере воспользоваться возможностями Faker и сделать данные более разнообразными, при генерации используются все (за исключением некоторых, например генератора csv или json) возможные методы генерации случайных данных.

В разработанный класс подается число таблиц N, которые нужно снегерировать, специфицируется язык для генерации контента (например, имен и городов, а также локализация учитывает специфичные особенности выбранного региона (это от самого функционала Faker)), диапазон в котором генерируются число столбцов (каждый раз случайно для каждой таблицы), и диапазон количества строк.

Итогом гереации является N сгенерированных .csv файлов, разнообразных настолько, насколько это позволяет функционал Faker.

Для исходной задачи создается два генератора, один для генерации в русской локализации, другой для генерации в английской.

```
user_name,mac_processor,hostname,street_address,mac_address,company_suffix,
feofan_1992,U; Intel,desktop-30.silovie.ru,"пр. Алтайский, д. 5",40:1f:c9:0
safonovanadezhda,U; Intel,db-47.rao.ru,"наб. Веселая, д. 13",ae:14:01:7d:66
aksenovaantonina,PPC,desktop-21.ao.biz,"бул. Металлистов, д. 57 стр. 738",3
naumovmartin,U; PPC,web-58.ip.com,"алл. Малиновая, д. 6/8 стр. 4/4",a4:b8:9
natan10,U; PPC,lt-65.ip.biz,"ул. Морская, д. 4/5",70:4f:be:07:f6:91,Групп,C
kozlovlavrenti,U; PPC,srv-41.npo.info,"ш. Шишкина, д. 3/9 стр. 235",74:bf:a
anisimovantip,U; Intel,desktop-02.fadeeva.edu,"наб. Медицинская, д. 9 стр.
mjasnikovvalentin,U; PPC,email-19.isaeva.edu,"пер. Казанский, д. 7 стр. 1",
ustinovvenedikt,U; Intel,laptop-06.ip.com,"ш. 50 лет ВЛКСМ, д. 1/4",54:3f:a
stojanchernov,Intel,srv-64.ip.ru,"алл. 70 лет Октября, д. 1",2a:90:f6:af:57
sinklitikija25,U; Intel,lt-94.rao.net,"алл. Крупской, д. 3 стр. 45",fa:a8:e
uorehov,U; PPC,laptop-83.ooo.ru,"бул. Светлый, д. 72",82:77:1e:6a:3e:d0,Инк
evdokimpanfilov,PPC,web-88.zao.edu,"пр. М.Горького, д. 7 к. 120",f4:dc:84:c
valentina03,PPC,db-00.rao.ru,"бул. Степана Разина, д. 4/6 стр. 6/4",46:46:f
taras54,Intel,lt-27.ooo.biz,"ш. Волгоградское, д. 4/9",fc:c4:a1:d6:b1:19,Ин
oleg_17,PPC,web-33.ooo.edu,"бул. Олимпийский, д. 9/5 к. 6/8",bc:b1:e3:a3:63
rjabovmodest,PPC,desktop-66.oao.org,"наб. Аэродромная, д. 6/3",84:b3:d8:56:
harlampi88,Intel,lt-96.makdonaldc.info,"бул. 9 мая, д. 319 стр. 45",7a:23:f
```

Рис. 1: Пример сгенерированных данных.

Использование реальных наборов данных

Так или иначе, функционал Faker ограничен, для расширения наборов данных было решено использовать общедоступные реальные датасеты.

Для поиска наборов данных использовалась платформа Kaggle, взаимодействие с которой велось через kaggle-api в Python.

По итогу разработан класс, который во-первых скачивает выбранное количество датасетов с использованием заранее заданных поисковых запросов (и фильтров на размер датасета и его типа (ищутся .csv)), а во-вторых совершает случайные запросы, и также скачивает, и распаковывает датасеты. Для каждого запроса он скачивает выбранное число датасетов (чтобы были схожие по тематике, но разные по данным наборы).

Заключение

По итогу, было использовано два разных подхода к получению данных. Первый, это генерация искусственных данных, такой подход может быть менее разнообразным, но его можно бесконечно расширять под разные реалии. Так, легко и гибко можно добавить самописный провайдер, который будет генерировать нужные нам, специфичные данные.

Второй подход, это использование реальных наборов данных. Такой подход менее контролируемый, но позволяет получать более раз-

нообразные данные.