# МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное автономное образовательное учреждение   
высшего образования

«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ   
АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ»

ИНСТИТУТ НЕПРЕРЫВНОГО И ДИСТАНЦИОННОГО ОБРАЗОВАНИЯ

|  |
| --- |
| КАФЕДРА 31 |

ОЦЕНКА

ПРЕПОДАВАТЕЛЬ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
| должность, уч. степень, звание |  | подпись, дата |  | инициалы, фамилия |

|  |
| --- |
| ОТЧЕТ О ПРОДЕЛАННОЙ РАБОТЕ |
|  |
|  |

РАБОТУ ВЫПОЛНИЛ

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| СТУДЕНТ ГР. | В0311 |  | |  |  | Истомин С.Н. |
|  | номер группы |  | | подпись, дата |  | инициалы, фамилия |
|  |  | |  | |  |  |

Санкт-Петербург 2016

# Введение

Количество перевозок авиационным транспортом непрерывно возрастает с момента появления авиации. По данным Международной Организации Гражданской Авиации (ICAO) количество только коммерческих рейсов в 2015 году достигло 33 000 000 рейсов.

По данным сайта FlightRadar.com количество бортов находящихся одновременно в мировом воздушном пространстве, не редко превышает 10000.

Не смотря на это количество акцидентов – инцидентов со смертельными исходами непрерывно снижается, начиная с середины семидесятых годов прошлого века. В частности, 2015 год стал одним из самых безопасных в истории авиации. К сожалению, 2016-ому уже не удастся повторить этот рекорд.

В связи с этим не перестает оставаться актуальным прогнозирование нештатных ситуаций

…. Кто и где исследует..

В этой работе, предпринята попытка на базе имеющихся открытых данных, как максимум создать нейро-сетевую модель предметной области по возможности, предсказывающую вероятности возникновения нештатных ситуаций, по тем или иным сочетаниям входных параметров. А как минимум провести анализ данных и выявить возможные корреляции классическими методами статистического анализа.

Источники исходных данных:

В качестве основы для построения модели решено было взять следующие данные:

* Базы данных инцидентов в авиации.
* Базы данных данных сводок погоды на аэродромах в формате METAR
* Статистика изменения ВВП по странам.
* Исторические данные о зонах военных конфликтов
* Базы данных координат аэродромов

В качестве источника данных по инцидентам был выбран сайт [https://aviation-safety.net](https://aviation-safety.net/) В связи стем что он содержит статистику ицидентов со всего мира кроме того..

…

Вышеназванные сайты содержат данные в сыром текстовом виде удобном для чтения но совершенно неудобном для обработки, в связи с чем первой задачей стало извлечение данных и подготовка их для использования в исследовании.

Для извлечения данных были написаны приложения-парсеры на языке программирования python с использованием библиотеки блегчающей написание парсеров – Scrapy.

При выборе представления данных для их использования в сети необходимо иметь в виду два принципа - количество переменных и содержательность данных . Из соображений сложности и скорости обработки часто бывает необходимо минимизировать количество переменных (входных и выходных) с которыми будет работать сеть. Проблемы также могут быть связаны с разрешением предъявляемых данных: чем выше разрешение, тем сложнее должна быть сеть и, следовательно, возрастает время тренировки и количество данных, необходимых для тренировки. Это приводит нас к второму аспекту представления тренировочных данных - ясность, содержательность. При ограничении количества входных переменных, возникает необходимость обеспечить, чтобы используемые переменные содержали информацию, достаточную для того, чтобы обеспечить обучение сети.

Для подготовки данных для использования в качестве обучающей выборки для обучения нейронной сети воспользуемся ритуалом описанным в книге Kevin Swingler "Applying Neural Networks. A practical Guide".

1. **Классификация данных по типу**. Каждый вход или выход нейронной сети может изменяться либо непрерывным образом, либо представлять набор дискретных значений. Тип каждой переменной необходимо идентифицировать, так как каждый тип требует своего подхода при кодировании.
2. **Вычисление глобальной статистики**. Для некоторых методов подготовки данных необходимо знать статистические свойства каждой переменной, используемой для тренировки. Для непрерывных данных должны быть вычислены среднее значение, стандартное отклонение, максимум и минимум. Для дискретных данных необходимо определить количество различающихся случаев (значений, которые может принимать каждая из переменных или категорий, из которых они выбираются).
3. **Удаление выбросов**. Девяносто пять процентов данных, распределенных по нормальному закону, находятся внутри интервала ограниченного удвоенным значением стандартного отклонения в окрестности среднего значения. Исключение из рассмотрения данных, находящихся вне указанного интервала является простейшим способом удаления выбросов. Удаление выбросов способствует более успешному обучению сети. Если же эти значения являются принципиальными для исследуемой задачи, тогда данные должны быть кодированы и собраны таким образом, чтобы правильно отобразить их важность.
4. **Количество тренировочных данных**. Существует множество факторов, влияющих на количество данных, необходимых для тренировки сети. Один из главных факторов - размерность данных. Чем больше переменных, тем больше данных требуется для тренировки сети. Эта проблема может быть решена либо увеличением набора данных, либо уменьшением их размерности.
5. **Построение выборки из набора данных**. Для очень больших наборов данных не всегда можно выполнить комплексный анализ на полном наборе. В таких случаях допустимо построить случайную выборку из данных, на основании которой и сделать заключения о методе предварительной обработки данных. Выборка должна быть случайной, должна включать данные со значениями из всего диапазона изменения и включать достаточно данных, чтобы быть репрезентативной. После построения выборки необходимо сравнить среднее значение и диапазон изменения с аналогичными параметрами полного набора, чтобы убедиться в том, что выборка выполнена правильно. Следующие процедуры могут быть выполнены на выборке, а не на полном наборе данных. Тем не менее сеть необходимо тренировать на полном наборе данных.
6. **Проверка качества**. Для построения хорошо сбалансированной модели важно, чтобы распределение тренировочных данных было гладким. Желательно также выполнить статистические тесты, чтобы убедиться в том, что данные содержат информацию, достаточную для решения поставленной задачи.
7. **Понижение размерности**. Нейронная сеть с большим количеством входных элементов содержит большое количество весовых коэффициентов (весов). Сеть с большим количеством весов требует (иногда экспоненциально) большого количества тренировочных данных. Аналогично, теория дискретизации утверждает, что имеет место экспоненциальная зависимость между размерностью данных и количеством точек дискретизации для корректного описания данных. Уменьшение количества входов сети позволяет использовать меньше данных, обеспечивая приемлемый уровень сложности сети.
8. **Шкалирование данных**. Выходные данные должны быть шкалированы (преобразованы) к диапазону, который соответствует диапазону выходных значений сжимающей функции активации выходного слоя. Сигмоидальная функция, например, имеет диапазон выходных значений от 0 до 1. Часто бывает удобным привести входные данные к тому же диапазону. Шкалирование может быть линейным или нелинейным, в зависимости от распределения данных. Наиболее часто используемые методы шкалирования - линейное, логарифмическое и "мягкое" (softmax). Можно также разделить данные на несколько диапазонов с различными факторами шкалирования.
9. **Кодирование данных**. Если данные были преобразованы, может возникнуть необходимость кодировать их перед предъявлением сети. Шкалированные численные данные могут быть использованы непосредственно, либо после дополнительного преобразования. Категорийные данные всегда необходимо кодировать.

# Этап сбора данных

Описание реализации парсера данных

Scrapy асинхронный фрэймворк для парсинга данных основанный на фрэймворке Twisted для языка Питон. В основе работы Twisted лежит шаблон проектирования «Реактор» который позволяет проводить асинхронную неблокирующую обработку http запросов.

Благодаря использованию этого паттерна, уменьшается время ожидания этому достигается высокая скорость работы Scrapy при анализе сайтов.

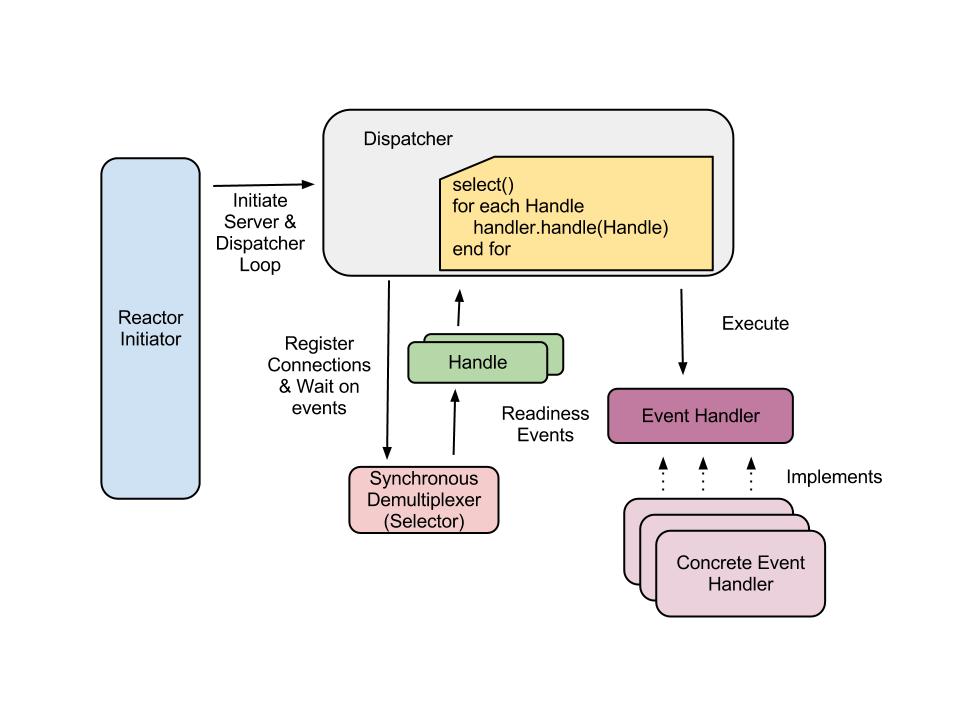
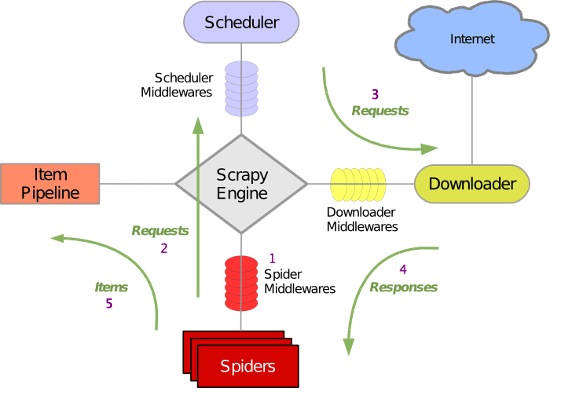
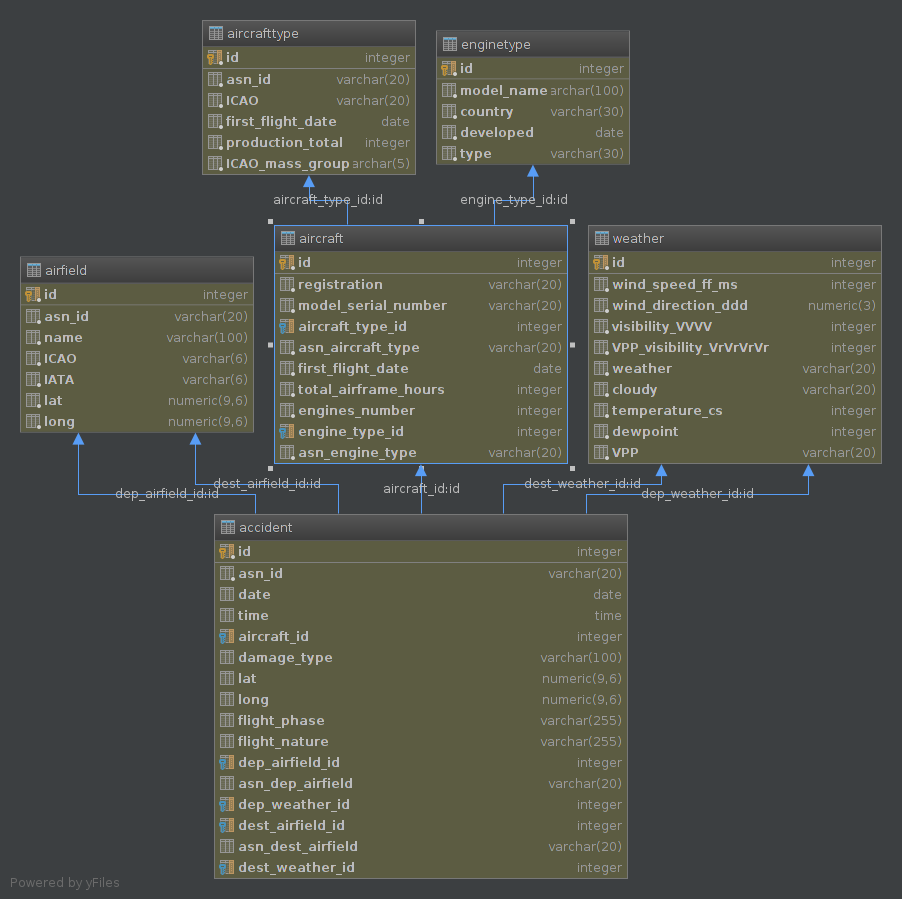


Схема обработки запроса в Scrapy.



Непосредствинная выемка данных из страницы происходит в конкретных реализациях класса Spider. Html страница рассматривается как XML документ (хотя строго говоря, это во многих случаях не так) и происходит последовательная выборка элементов при помощи язык запросов к элементам [XML](https://ru.wikipedia.org/wiki/XML)-документа - XPath с последующей дообработкой при помощи регулярных выражений.

Полученные данные сохранялись в базу данных Postgres при помощи ORM фрэймворка peewee. Получившаяся структура таблиц показана на следующей схеме ниже.



Файлы проекта опубликованы на сервисе github и доступны по адресу: <https://github.com/Neanderthal/Aviation_Accidents_Prediction>