

**Оценочные материалы дополнительной профессиональной программы
повышения квалификации
«Разработка нейросетей на Python с нуля до проекта (вторая ступень)»**

Пример оценочных материалов для текущего контроля

1. (SC) Что является основной целью EDA перед моделированием?

- A. Подбор гиперпараметров модели
- B. Предварительное исследование данных для выявления структуры, проблем и закономерностей
- C. Увеличение выборки за счёт аугментаций
- D. Снижение времени обучения модели

Ответ: B. Предварительное исследование данных для выявления структуры, проблем и закономерностей.

2. (MC) Какие задачи относятся к целям EDA?

- A. Понимание структуры и характеристик данных
- B. Выявление проблем качества (пропуски, выбросы)
- C. Гарантированное повышение точности любой модели
- D. Формирование гипотез о взаимосвязях переменных
- E. Выбор информативных признаков
(выберите все подходящие)

Ответ: A, B, D, E.

A. Понимание структуры и характеристик данных; B. Выявление проблем качества; D. Формирование гипотез; E. Выбор информативных признаков.

3. (ORD) Расположите этапы анализа данных в корректной последовательности: 1 — Анализ данных; 2 — Интерпретация результатов; 3 — Сбор данных; 4 — Подготовка данных.

Введите порядок номеров: _____

Ответ: 3 → 4 → 1 → 2.

4. (MATCH) Соотнесите инструмент и назначение:

- 1. Jupyter Notebook → a) Численные операции с массивами
- 2. pandas → b) Визуализация графиков
- 3. NumPy → c) Интерактивная среда разработки/документирования
- 4. Matplotlib/Seaborn → d) Обработка табличных данных
- 5. Укажите соответствия в формате «1—?, 2—?, 3—?, 4—?».

Ответ: 1—c, 2—d, 3—a, 4—b.

5. (SC) Что делает data.head() в pandas?

- A. Строит гистограмму первого столбца
- B. Выводит размеры таблицы
- C. Показывает первые строки датафрейма
- D. Удаляет пропуски из набора данных

Ответ: С. Показывает первые строки датафрейма.

6. (MC) Какие показатели относятся к мерам разброса (вариативности)?

- A. Среднее
- B. Медиана
- C. Дисперсия
- D. Стандартное отклонение
- E. Мода

(выберите все подходящие)

Ответ: C, D.

C. Дисперсия; D. Стандартное отклонение.

7. (SC) Какой график уместен для зависимости между двумя количественными переменными?

- A. Гистограмма
- B. Диаграмма рассеяния (scatter plot)
- C. Круговая диаграмма
- D. Ящик с усами (box-plot)

Ответ: B. Диаграмма рассеяния (scatter plot).

8. (MC) Какие типичные ошибки новичков описаны в материале?

- A. Игнорирование проверки и подготовки данных
- B. Недооценка статистики и отказ от визуализаций
- C. Слишком ранний выбор простых моделей вместо сложных
- D. Пере усложнение моделей без необходимости
- E. Отсутствие документирования шагов анализа

(выберите все подходящие)

Ответ: A, B, D, E.

A. Игнорирование проверки/подготовки данных; B. Недооценка статистики и отказ от визуализаций; D. Пере усложнение моделей; E. Отсутствие документирования.

9. (MC) Какие признаки чаще всего признаются ненужными и удаляются?

- A. Постоянные (категориальные) признаки с одним значением
- B. Признаки с высокой долей пропусков
- C. Уникальные идентификаторы строк (случайные ID)
- D. Целевая переменная
- E. Сильно коррелирующие между собой признаки (мультиколлинеарность)

(выберите все подходящие)

Ответ: A, B, C, E.

A. Постоянные категориальные; B. С высокой долей пропусков; C. Уникальные ID; E. Сильно коррелирующие между собой.

10. (TF) Верно или неверно: «EDA помогает обнаружить дисбаланс классов и выбрать корректные стратегии до обучения модели».

- A. Верно

- B. Неверно

Ответ: A. Верно.

Пример оценочных материалов для промежуточной аттестации

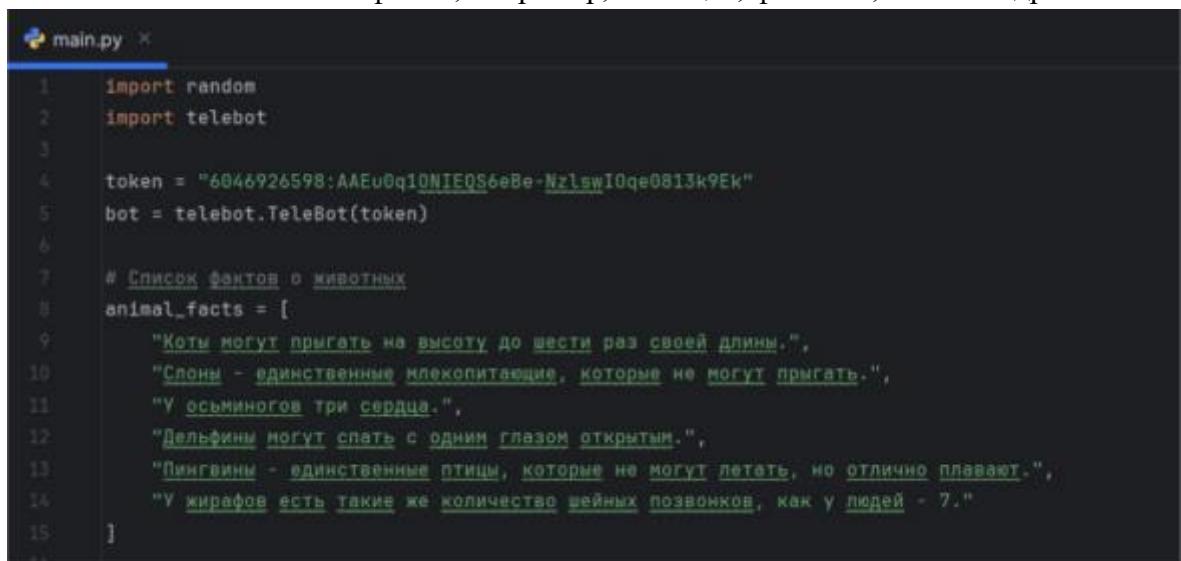
Создание Telegram-бота для генерации случайных фактов

В рамках этого задания мы поработаем со списками и циклом for. В качестве задания необходимо будет создать Telegram-бот, который будет отправлять пользователю случайные факты о животных. Пользователь может запрашивать факты, и бот будет отвечать на каждую команду. Бот должен использовать цикл for для отправки нескольких фактов за один запрос.

Итак, в рамках этого задания мы будем использовать бот, который ранее создавали на втором задании. Если Вы пропустили этот шаг, то можно вернуться ко второму заданию и вспомнить как создать через BotFather свой собственный бот.

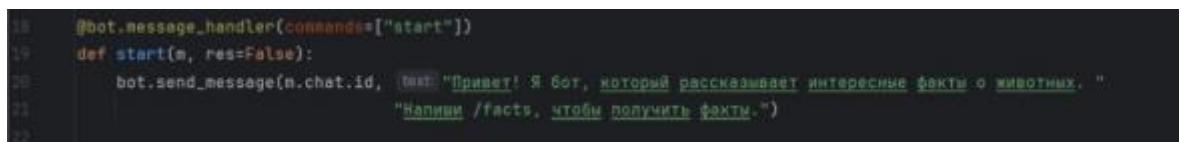
Факты о животных мы будем хранить в списке. Создадим этот список и добавим туда случайные факты о животных.

Вы можете написать свои факты, например, о птицах, фильмах, книгах и др.



```
main.py
1 import random
2 import telebot
3
4 token = "6046926598:AAEu0q10NIEQS6eBe-NzlsWI0qe0813k9Ek"
5 bot = telebot.TeleBot(token)
6
7 # Список фактов о животных
8 animal_facts = [
9     "Коты могут прыгать на высоту до шести раз своей длины.",
10    "Слоны - единственное млекопитающее, которые не могут прыгать.",
11    "У осьминогов три сердца.",
12    "Дельфины могут спать с одним глазом открытым.",
13    "Пингвины - единственные птицы, которые не могут летать, но отлично плавают.",
14    "У жирафов есть такие же количество шейных позвонков, как у людей - 7."
15]
```

Реализуйте команду /start, которая будет отправлять пользователю приветственное сообщение:



```
1 @bot.message_handler(commands=["start"])
2 def start(m, res=False):
3     bot.send_message(m.chat.id, text="Привет! Я бот, который рассказывает интересные факты о животных.\nНапиши /facts, чтобы получить факты.")
```

Далее, реализуйте команду /facts, которая будет отправлять пользователю 3 случайных факта.

Так как по заданию количество фактов, отправляемых пользователю известно заранее, то нам необходимо использовать цикл for для перебора и отправки фактов. Генерацию случайных фактов выполните через команду random.sample (animal_facts, 3).

```
24 @bot.message_handler(commands=["facts"])
25 def facts(m, res=False):
26     # Получаем 3 случайных факта
27     selected_facts = random.sample(animal_facts, k=3)
28     # Отправляем факты пользователю
29     for fact in selected_facts:
30         bot.send_message(m.chat.id, fact)
31
32
33 bot.polling(none_stop=True, interval=0)
```

Ваш бот готов!

Не забудьте запустить его на выполнение через команду polling!

Вы можете придумать свой собственный бот, который будет хранить список и выдавать пользователю по запросу информацию из этого списка либо выполнить задание, которое мы рассмотрели на уроке.

Пример оценочных материалов для итоговой аттестации

Методические указания к подготовке итоговой аттестационной работе

Основная цель итоговой аттестационной работы – выполнить работу, демонстрирующую уровень подготовленности к самостоятельной профессиональной деятельности.

Процедура защиты итоговой аттестационной работы

ИАР может выполняться как индивидуально, так и в группах 2-3 человека. Защита ИАР включает презентацию работы перед комиссией. Защита ИАР дает возможность продемонстрировать уровень приобретенных слушателем профессиональных компетенций.

Слушатель предоставляет результат выполненной работы в формате PDF, оформленной и отвечающей требованиям к содержанию итоговой аттестационной работы. Список использованных источников литературы приводится в конце ИАР.

Объем презентации следует выбирать исходя из длительности выступления (обычно — не более 5–7 минут). В выступлении должны быть четко обозначены область и актуальность работы, постановка задачи, приведены результаты, полученные слушателем.

Защита ИАР может проходить как в синхронном формате в режиме видеоконференцсвязи, так и в асинхронном формате, когда слушатель или группа слушателей записывают свой доклад по проделанной работе и прикрепляют его в электронный курс вместе с текстом ИАР.

Требования к устному докладу

1. Приветствие, обращение к членам комиссии и представление.
2. Тема итоговой аттестационной работы.
3. Актуальность, цель и задачи работы.
4. Анализ результатов работы.
5. Заключение.

Продолжительность выступления – 7–8 минут.

По результатам защиты ИАР аттестационная комиссия принимает решение о предоставлении слушателям по результатам освоения дополнительной профессиональной программы профессиональной переподготовки права заниматься профессиональной деятельностью в сфере разработки программного обеспечения и выдаче диплома о профессиональной переподготовке.

Примерные темы ИАР

Примерные темы итоговых аттестационных работ:

1. Классификация изображений с использованием сверточных нейронных сетей.
2. Прогнозирование спроса или временных рядов с помощью рекуррентных архитектур (LSTM, GRU).
3. Обнаружение объектов на изображениях с использованием TensorFlow/Keras.

4. Сравнение архитектур CNN и MLP на задаче классификации табличных данных.

5. Оптимизация нейросетевой модели для работы в ограниченной вычислительной среде (MVP).

6. Применение transfer learning для решения прикладной задачи компьютерного зрения.

Обучающиеся предоставляется право самостоятельно определить тему ИАР.

Структура и содержание итоговой аттестационной работы

Задачи итоговой аттестационной работы должны иметь общетеоретическое или практическое значение.

Структура:

Введение:

- Описание задачи. Актуальность.
- Описание подхода к разработке приложения на языке Python для решения задачи.
- Цель работы.
- Задачи.

1. Обзорная часть

Название обзорной части можно сделать более специфическим, например, «Обзор алгоритмов анализа данных», в зависимости от типа разрабатываемого приложения.

1.1. Обзор задачи

Название также можно сделать более специфическим, например, «Обзор задачи создания web-приложений». В этом разделе необходимо привести постановку задачи. Специфику разработки выбранного типа приложения, краткую историю развития технологий.

1.2. Обзор инструментов для создания приложений

В разделе приводится краткое описание существующих инструментов для создания выбранного типа приложений. Для задач машинного обучения можно привести описание библиотек Pandas и NumPy.

2. Проектирование приложения

2.1. Описание желаемых функций приложения

В общем и целом, для рассматриваемого продукта необходимо сформулировать функциональные и нефункциональные требования.

2.2 Требования к программе или программному изделию

В разделе «Требования к программе или программному изделию» могут содержаться следующие нефункциональные требования:

- требования к надежности;
- требования к составу и параметрам технических средств;
- требования к информационной и программной совместимости;

Нефункциональное требование – это описание присущих свойств или характеристик, которые система должна демонстрировать, или ограничения, которые она должна соблюдать, требования к качеству и так далее.

При формулировании требований к качеству необходимо рассмотреть следующие:

- атрибуты внешнего качества: производительность, надёжность, доступность, масштабируемость, безопасность;

- атрибуты качества в использовании: результативность, эффективность, скорость обучения, точность, утомляемость;

- ограничения: совместимость с оборудованием, системным ПО, ограничения на языки и средства разработки, допущения о квалификации пользователей, состав поставки, лицензионные ограничения.

2.3 Стадии и этапы разработки

В разделе «Стадии и этапы разработки» устанавливают необходимые стадии разработки, этапы и содержание работ (перечень программных документов, которые должны быть разработаны, согласованы и утверждены, если такие предусмотрены).

Например:

Разработка должна быть проведена в три стадии:

- 1) разработка технического задания;
- 2) рабочее проектирование: дизайн, создание контента, разработка
- 3) Тестирование, обзор и запуск
- 4) Поддержка.

Также здесь приводится подробное описание стадий разработки.

2.4 Описание архитектуры приложения

Описание основных компонентов приложения и их взаимодействия:

- Основные модули программы
- Используемый инструмент для создания приложения (Google Colab, Jupyter Notebook и т.д.).

3. Реализация приложения

3.1 Написание программного кода

Приводятся основные фрагменты кода приложения.

3.2 Тестирование приложения

Приводятся примеры использования приложения для определения его работоспособности. Например, скриншоты с результатами распознавания. Нужно загрузить несколько изображений и проанализировать результаты (все ли заявленные функции приложения работают).

Заключение:

Краткое описание полученных результатов:

- Описание задачи
- Описание инструмента создания приложения
- Описание приложения.
- Планируемые дальнейшие шаги.

Порядок выполнения итоговой аттестационной работы

Этапы выполнения ИАР задаются графиком выполнения и контролируются руководителем:

1. Выбор и согласование с руководителем темы ИАР работы.
2. Формулировка цели и задач ИАР работы. Обоснование актуальности выбранного направления работы.
3. Литературно-патентный поиск по тематике работы. Поиск аналогичных решений. Формирование концепции предлагаемых подходов, методов.

4. Разработка методов достижения цели проекта. Выбор и/или проектирование математического, программного, технического и иных видов обеспечений для достижения цели и решения задач ИАР.
5. Моделирование. Разработка программной или программно-аппаратной реализации проекта.
6. Проведение практических испытаний и анализ результатов.
7. Обобщение проведенных исследований. Формирование итоговой аттестационной работы.
8. Публичная защита ИАР.