**Технологии программирования для NLP**

**Блок 1**

1. Дайте определение технологии обработки естественных языков (NLP), опишите в каких областях она широко используется в настоящее время.
2. Опишите основные этапы и основные операции по обработке текстовых данных в области обработки естественных языков (NLP).
3. Опишите основные концепции контекстно-свободной грамматики и методов исчисления предикатов.
4. Дайте характеристику основным этапам обработки текстов в NLP, нарисуйте схему этапов обработки текстов.
5. Приведите характеристику различных методов векторизации текстов, таких как Bag of words, Tf-idf и другие.
6. Опишите векторизацию текстов с помощью word embeddings: Word2Vec, FastText и т.д.
7. Дайте подробное описание этапов обучения модели классификации текстов. Опишите алгоритмы классификации Naïve Bayes и Logistic Regression.
8. Опишите этапы классификации текстов. Охарактеризуйте алгоритмы классификации Support Vector Machine и Метод k-ближайших соседей.
9. Дайте характеристику методам классификации текстов. Опишите алгоритмы классификации Decision Tree и Random Forest.
10. Опишите этапы классификации текстов. Охарактеризуйте алгоритмы классификации XgBoost и CatBoost.
11. Опишите, какие типы нейронных сетей обычно используются для классификации текстов.
12. Дайте характеристику нейронной сети прямого распространения (Feedforward neural network).
13. Опишите применением сверточных нейронных сетей (CNN) в области NLP для классификации текстов.
14. Объясните, как рекуррентные нейронные сети (RNN) могут быть использованы для классификации текстов.
15. Дайте характеристику нейронной сети LSTM (Long Short-Term Memory) и почему они эффективны в задачах классификации текстов.
16. Объясните преимущество использования GRU в контексте NLP.
17. Объясните принцип работы и преимущества использования сетей Transformer для классификации текстов.
18. Охарактеризуйте модель BERT (Bidirectional Encoder Representations from Transformers) для классификации текстов.
19. Опишите модель RoBERTa для классификации текстов, и в чем есть ее отличие от модели BERT.
20. Опишите различия между одномерными и двумерными сверточными нейронными сетями при работе с текстовыми данными.

**Блок 2**

1. Опишите, какие методы обработки текстов необходимо использовать перед применением нейронных сетей для классификации текстов.
2. Опишите стратегии, которые используются для борьбы с переобучением нейронных сетей в задачах классификации текстов.
3. Объясните применение подхода Transfer Learning в классификации текстов с использованием нейронных сетей.
4. Приведите методы балансировки в задачах классификации текстов.
5. Опишите основные характеристики метода балансировки классов RandomOversampling.
6. Опишите подробно метод балансировки классов SMOTE.
7. Дайте подробную характеристику GPT (Generative Pre-trained Transformer), и как он используется для анализа текстов.
8. Опишите сновные отличия между моделями GPT и GPT-2 в контексте анализа текстов.
9. Охарактеризуйте улучшение возможностей GPT-3 по анализу текста по сравнению с предыдущими версиями.
10. Объясните, как можно использовать ChatGPT для автоматической генерации ответов на базе анализа текстов.
11. Опишите ограничения и недостатки моделей GPT при работе с специализированными или узконаправленными текстами.
12. Опишите основные метрики для оценки производительности моделей классификации текста с использованием нейронных сетей.
13. Опишите подробно парсинг текстов и задачи, для которых он используется.
14. Дайте характеристику основным функциям и возможностям библиотеки BeautifulSoup для парсинга HTML и XML документов?
15. Опишите отличия между BeautifulSoup и Scrapy с точки зрения подходов к парсингу данных.
16. Объясните применение Selenium WebDriver для извлечения данных с веб-страниц, которые требуют взаимодействия с JavaScript.
17. Опишите наиболее полезные методы Selenium WebDriver для автоматизации сбора данных.
18. Объясните, как Selenium можно интегрировать с другими библиотеками Python для улучшения процесса парсинга данных.
19. Опишите подходы к управлению данными и их хранению при использовании Scrapy.
20. Охарактеризуйте проблемы безопасности при парсинге данных.

**3 блок**

1. Создайте небольшой скрипт по загрузке данных с помощью библиотеки Pandas и очистке данных от лишних символов с помощью регулярных выражений.
2. Создайте небольшой скрипт на Python с добавлением стемминга слов.
3. Создайте скрипт на Python с применением лемматизации слов.
4. Создайте скрипт использования one-hot encoding для категориальных переменных с помощью Pandas или Scikit-learn.
5. Создайте скрипт по работе с регулярными выражениями в Python для очистки текстовых данных.
6. Напишите скрипт векторизации текстовых данных с помощью метрики *Tf-idf* на Python.
7. Создайте скрипт на Pandas для загрузки нескольких данных и их слияния в единый DataFrame.
8. Создайте скрипт по предобработке текстовых данных и их разделения на train и test части.
9. Создайте скрипт масштабирования данных в библиотеке Scikit-learn.
10. Создай небольшой скрипт классификации текстов данных с помощью библиотеки Scikit-learn.
11. Напишите небольшой скрипт векторизации с помощью Bag of words на Python
12. Дан небольшой скрипт. Дополни его команда для классификации с помощью Decision Tree.

import tensorflow as tf

from tensorflow.keras.preprocessing.text import Tokenizer

from tensorflow.keras.preprocessing.sequence import pad\_sequences

from tensorflow.keras.models import Sequential

from tensorflow.keras.layers import Dense, Embedding, GlobalAveragePooling1D

# Пример данных

reviews = [

 "Great movie! Must watch.",

 "Worst movie ever.",

 "It was a fantastic performance!",

 "Not a good movie, boring plot.",

 "The cinematography is pretty great.",

 "I hated the storyline."

]

labels = [1, 0, 1, 0, 1, 0] # 1 - положительный, 0 - отрицательный

1. Дан небольшой скрипт. Дополни его команда для классификации с помощью Random Forest.

import tensorflow as tf

from tensorflow.keras.preprocessing.text import Tokenizer

from tensorflow.keras.preprocessing.sequence import pad\_sequences

from tensorflow.keras.models import Sequential

from tensorflow.keras.layers import Dense, Embedding, GlobalAveragePooling1D

# Пример данных

reviews = [

 "Great movie! Must watch.",

 "Worst movie ever.",

 "It was a fantastic performance!",

 "Not a good movie, boring plot.",

 "The cinematography is pretty great.",

 "I hated the storyline."

]

labels = [1, 0, 1, 0, 1, 0] # 1 - положительный, 0 - отрицательный

1. Дан небольшой скрипт. Дополни его команда для классификации с помощью Dense neural network

import tensorflow as tf

from tensorflow.keras.preprocessing.text import Tokenizer

from tensorflow.keras.preprocessing.sequence import pad\_sequences

from tensorflow.keras.models import Sequential

from tensorflow.keras.layers import Dense, Embedding, GlobalAveragePooling1D

# Пример данных

reviews = [

 "Great movie! Must watch.",

 "Worst movie ever.",

 "It was a fantastic performance!",

 "Not a good movie, boring plot.",

 "The cinematography is pretty great.",

 "I hated the storyline."

]

labels = [1, 0, 1, 0, 1, 0] # 1 - положительный, 0 - отрицательный

1. Дан небольшой скрипт. Дополни его команда для классификации с помощью Long short-term memory neural network

import tensorflow as tf

from tensorflow.keras.preprocessing.text import Tokenizer

from tensorflow.keras.preprocessing.sequence import pad\_sequences

from tensorflow.keras.models import Sequential

from tensorflow.keras.layers import Dense, Embedding, GlobalAveragePooling1D

# Пример данных

reviews = [

 "Great movie! Must watch.",

 "Worst movie ever.",

 "It was a fantastic performance!",

 "Not a good movie, boring plot.",

 "The cinematography is pretty great.",

 "I hated the storyline."

]

labels = [1, 0, 1, 0, 1, 0] # 1 - положительный, 0 - отрицательный

1. **Напишите функцию на Python, использующую BeautifulSoup, для извлечения всех заголовков (тегов <h1>) с веб-страницы.**
2. **Напишите пример кода на Python для чтения JSON файла и извлечения данных из него.**
3. **Приведите пример кода на Python для парсинга данных из таблицы на веб-странице и сохранения этих данных в формате CSV с использованием pandas.**
4. **Опишите процесс использования XPath с библиотекой lxml для извлечения данных из HTML.**
5. **Напишите код на Python с использованием Scrapy, который собирает заголовки и URLы всех статей в блоге по заданной теме.**