**Курсовая работа по дисциплине “Технологии анализа данных и машинное обучение“**

**Цель работы**

Продемонстрировать владение основными навыками обучения и применения моделей глубокого обучения, владение основными инструментальными средствами библиотек языка программирования Python для глубокого обучения, в том числе фреймворком для глубокого обучения PyTorch, методами сбора и подготовки набора данных, анализа качества обучения модели глубокого обучения, умение делать выводы из проведенного анализа.

**Задачи для выполнения**

1. Выбрать (собрать) набор данных для анализа в соответствии с выбранной темой курсовой работы. Описать этот набор и решаемую задачу. Определить метрики качества для модели, определить типичный уровень метрик для решения аналогичных задач, установить ориентиры для метрики.
2. Провести предварительный анализ и очистку данных. Этот этап включает в себя вывод информации о количественных характеристиках датасета, информацию об отсутствующих значениях, характеристиках и физическом смысле каждого атрибута данных, его значимости для предсказания целевой переменной, вывод нескольких элементов данных для иллюстрации структуры данных.
3. При необходимости выполнить преобразование данных. Этот этап сильно зависит от типа исследуемых данных и может включать в себя токинизацию и векторизацию текста, извлечение признаков из данных, преобразование изображений в плоский численный массив и другие преобразования.
4. Разделить набор данных на обучающую, тестовую и валидационную выборки. Обосновать количественные характеристики и метод разделения (временной, случайный, последовательный) выборки.
5. Определить принципиальную архитектуру нейросетевой модели глубокого обучения, включая функцию ошибки и методы регуляризации, общую логику обучения модели. Предусмотреть возможные варианты модификации модели.
6. Реализовать модель с помощью фреймворка PyTorch. Реализация должна включать функционал загрузки данных, вывода промежуточных и финальных метрик качества и диагностических параметров процесса обучения модели. Реализация должна предусматривать гибкость, позволяющую выполнять ограниченную модификацию модели для выбора наилучшей из альтернативных архитектур.
7. Настроить параметры входных данных и параметров для обучения. Провести обучение модели на подготовленных входных данных используя выбранные настройки и алгоритмы обучения.
8. Провести оценку качества модели с использованием предусмотренных метрик качества, проверить модель на переобучение.
9. Провесит обучение и оценку качества для несколько альтернативных архитектур (вариаций архитектур) моделей (не менее 3) для решения выбранной задачи. Проанализировать результаты, сделать выводы, в том числе определить наилучшую архитектуру.
10. Выполнить тонкую настройку модели с помощью подбора значений гиперпараметров. Провести подбор не менее трех гиперпараметров, при этом как минмум для одного подбор значения параметра выполнить в автоматическом режиме (с помощью Grid Search или аналогичных методов).
11. Представить результаты обучения модели в наглядном виде (графики, линии обучения, таблицы сравнения моделей, таблицы классификации, и другие). Сделать выводы, сравнить с существующими аналогичными решениями, порассуждать о перспективах улучшения методов решения проблемы.

В зависимости от формулировки выбранной темы, объем и наличие пунктов их этого списка может варьироваться. Пункты 1,3,11,9 являются обязательными для всех тем курсовых работ.

**Методические указания**

1. Работа выполняется в виде программного ноутбука Python Jupyter. Пояснительная записка выполняется в виде текстового документа и должна включать в себя: титульный лист, текстовое описание проблемы, ссылку на публично доступный репозиторий с полным кодом выполнения работы, по необходимости пример кода для каждого этапа работы, текстовые выводы по каждому этапу и сформулированное заключение с результатами работы и их интерпретация.
2. Все пояснения, выводы и замечания, на которые необходимо обратить внимание должны присутствовать в работе в виде ячеек документации либо (менее предпочтительно) программных комментариев.
3. Работа должна выполняться студентом самостоятельно и индивидуально.
4. Оценка качества моделирования должна производиться с использованием определенных метрик. Их выбор должен быть описан и обоснован до начала моделирования. Плюсом работы является широкий набор метрик эффективности моделей.
5. Отчет работы производится в формате презентации. Слушатели (включая преподавателя) могут задавать вопросы представляющему свою работу студенту. Регламент презентации - 5 минут на выступление, 2 минуты на вопросы.

**Критерии оценки**

1. Структурированность отчета. В работе должна прослеживаться четкая структура - подготовительный этап, анализ данных, построение простых моделей, сравнение и анализ моделей, выводы, построение моделей с учетом выводов, итоговый результат.
2. Наличие выводов. Работа должна содержать текстовые замечания, поясняющие каждый шаг работы студента: что делается, зачем и какую информацию это нам дает. Оценивается полнота и адекватность выводов.
3. Визуализация. Работа должна демонстрировать навыки студента визуализировать информацию. Особенно на этапах описательного анализа и анализа обучаемости модели. Оценивается разнообразие, наглядность и информативность визуализации.
4. (в зависимости от темы) Разнообразие моделей. Студент должен продемонстрировать умение работать с разнообразными моделями обучения, применимыми к одной задаче. Например, в задачах классификации существует как минимум десять наиболее применимых моделей. Оценивается число алгоритмов, примененных студентом для одной и той же задачи.
5. (в зависимости от темы) Улучшение модели. Студент должен продемонстрировать умение анализировать обученную модель и искать пути для ее совершенствования. Оценивается количество итераций совершенствования модели и их эффективность.
6. Предобработка данных. Работа должна содержать исчерпывающий алгоритм предварительной обработки данных. Он служит для того, чтобы исправить все несовершенства в данных и сделать набор данных как можно более пригодным для машинного обучения. Оценивается сложность и воспроизводимость процедуры предварительной обработки данных.
7. Использование метрик эффективности и оценка валидности результатов. Оценивается разнообразие и адекватность задаче примененных метрик эффективности (включая время обучения); корректность проверки модели на переобучение; полнота сравнения и правильность выводов из сравнения моделей по разным метрикам.

**Сроки выполнения**

Указаны в соответствующем разделе сайта департамента: <http://www.fa.ru/org/dep/findata/Pages/kurs.aspx> .

Пример титульного листа пояснительной записки:

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ БЮДЖЕТНОЕ

УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

**«ФИНАНСОВЫЙ УНИВЕРСИТЕТ ПРИ ПРАВИТЕЛЬСТВЕ**

**РОССИЙСКОЙ ФЕДЕРАЦИИ»**

**(ФИНАНСОВЫЙ УНИВЕРСИТЕТ)**

Департамент анализа данных

и машинного обучения

***Дисциплина: «Технологии анализа данных и машинного обучения»***

*Направление подготовки: «Прикладная математика и информатика»*

*Профиль: «Анализ данных и принятие решений в экономике и финансах»*

*Факультет информационных технологий и анализа больших данных*

*Форма обучения очная*

*Учебный 2022/2023 год, 7 семестр*

**Курсовая работа на тему:**

«Тема курсовой работы»

*Выполнил(а):*

студент(ка) группы ПМ19-1

Лашуков Т. Д.

*Научный руководитель:*

доцент к.э.н. Макрушин С. В.

**Москва 2022**

Список литературы:

Основная литература:

1. М.В. Коротеев. Об основных задачах дескриптивного анализа данных.
2. М.В. Коротеев. Учебное пособие по дисциплине “Анализ данных и машинное обучение” - 2018.

Дополнительная литература

1. A. Geron. Hand on Machine Learning with scikit-learn and Tensorflow - 2017 (564p)
2. C. Albon. Machine learning with Python Handbook - 2018 (427p)
3. L.P. Coelho, W. Richert. Building machine learning systems with Python - 2015 (326p)
4. J. Grus. Data science from scratch - 2015 (330p)
5. W. McKiney. Pandas: powerful Python data analysis toolkit - 2016 (1971p)