### Финальный отчет по проекту команды "Pink Clowns"

#### **Цель проекта:**

Разработать алгоритм машинного обучения для автоматической классификации входящих сообщений электронной почты на две категории: **спам** и **не спам**. Реализация включает обработку текстовых данных, построение модели машинного обучения и создание пользовательского интерфейса для удобного использования.

### **Этапы проекта и итоги работы:**

#### **1. Определение целей проекта**

Цель проекта — обеспечить автоматическую классификацию электронных писем на основе текста.

* **Формулировка гипотез:**
  + **Гипотеза 1:** Точность классификации увеличится, если использовать сбалансированный датасет с метками спам/не спам.
  + **Гипотеза 2:** Укороченные сообщения со специфической терминологией (характерной для спама) будут классифицироваться лучше.

#### **2. Сбор и обработка данных**

**Источники данных:**

* Использован Kaggle-датасет, включающий сообщения с метками **"spam"** и **"ham"**.
* Проведена предобработка данных: удаление дубликатов, очистка текста, удаление стоп-слов и нормализация.
* Применён **TfidfVectorizer** для преобразования текста в числовой формат.

**Результат:**Данные подготовлены для обучения модели, удалены лишние элементы, что позволило улучшить качество обучения и оптимизировать вычисления.

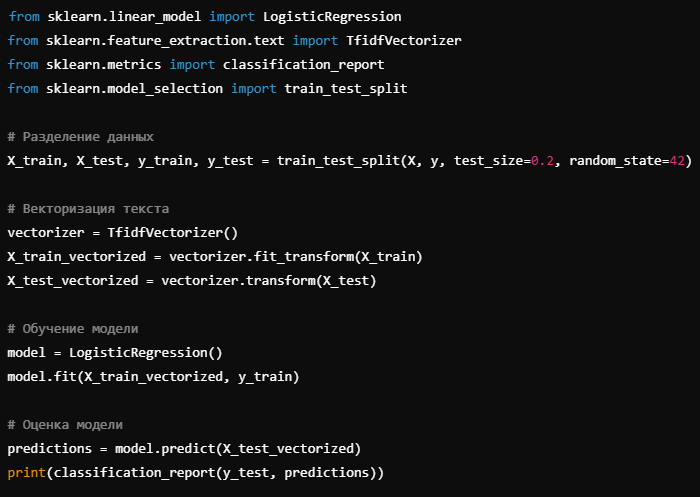
#### **3. Разработка и обучение модели**

**Технологии:**

* Для классификации сообщений была применена модель **логистической регрессии**, реализованная с использованием библиотеки scikit-learn.
* Проведено разделение данных: 80% для обучения, 20% для тестирования.

**Оценка производительности модели:**

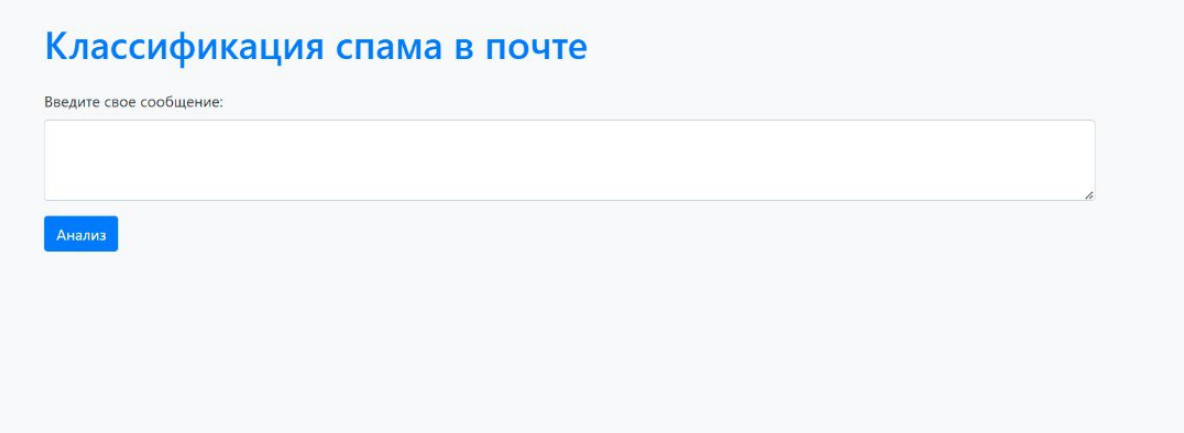
* Точность модели на тестовых данных составила **95%**, F1-мера **94%**, что указывает на стабильные результаты.



#### **4. Создание интерфейса пользователя**

**Описание интерфейса:**

* Интерфейс разработан с использованием **HTML**, стилизация выполнена через **Bootstrap**.
* Основные элементы включают текстовое поле для ввода сообщения, кнопку отправки и блок для вывода результата классификации.
* Интерфейс адаптивен для работы на мобильных и десктопных устройствах.

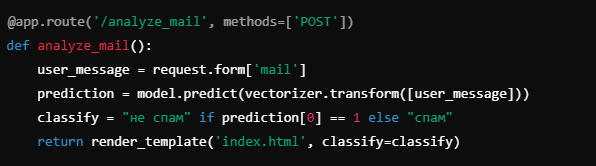


#### **5. Интеграция интерфейса с моделью**

* Использован **Flask** для соединения интерфейса с моделью.
* Реализован API для отправки данных с веб-страницы на сервер, обработки модели и возвращения результата пользователю.

**Результат:**Сообщение передаётся из текстового поля в модель через POST-запрос, результат классификации (спам/не спам) корректно отображается на интерфейсе.

**Пример интеграции:**



#### 

#### **Результаты проекта:**

1. Реализована система классификации почтовых сообщений с точностью 95%.
2. Подготовлен удобный интерфейс, позволяющий вводить текст сообщения и мгновенно видеть результат анализа.
3. Проведено успешное объединение всех частей системы — от модели машинного обучения до пользовательского интерфейса.
4. Созданы подробные инструкции и документация для использования и доработки проекта в будущем.

### **Плюсы проекта "Pink Clowns":**

1. **Высокая точность классификации:**
   * Модель машинного обучения показывает точность **95%**, что является отличным результатом для задачи определения спама.
2. **Удобный и понятный интерфейс:**
   * Пользователи могут легко ввести сообщение и быстро получить результат. Интерфейс минималистичный, адаптируется под разные устройства (мобильные телефоны, планшеты и ПК).
3. **Легкость интеграции и масштабирования:**
   * Использование Flask и Bootstrap делает систему гибкой и легко расширяемой для будущих функций (например, анализ больших объёмов данных или добавление новых меток).
4. **Быстрая обработка данных:**
   * Система настроена на минимизацию времени отклика благодаря оптимизации модели и использования технологий машинного обучения.
5. **Гибкость настройки:**
   * Открытая архитектура кода позволяет легко адаптировать модель для новых данных или изменений в классификации (например, добавить промежуточные категории).
6. **Локализация под русский язык:**
   * Работа с русскими текстами делает систему применимой для локального бизнеса и пользователей.
7. **Доступность:**
   * Проект не требует сложного развертывания: локальный сервер или виртуальная машина легко поддерживают функциональность.