**Идеи для проектов**

1. Симуляция почерка при помощи нейросети.

2. Парсинг с выводом в excel.

3. Датчик движения с выводом сообщения в телеграмм.

4. Тоже самое, но при помощи камеры с распознаванием людей и животных.

5. Нейросеть для генерации рецептов, исходя из фото продуктов в холодильнике.

6. Разработка программ с интерфейсом и exe-файлом.

7. Светодиодный куб 4х4х4

8. Распознавание дефектов на поверхности горячего проката. Ядро – CNN, имеется код для обучения, для работы на ПК и код графического интерфейса

9. Пример работы ШИМ и АМ-генераторов. Можно расширить функционал и сделать графический интерфейс, отлично подходить для дипломной работы в техникуме.

10. Пример расчёта ипотеки, код также готов, только расширение функционала, интерфейс, бэкэнд для телеграмм-бота и т.п.

11. Простой код на криптографию – код Хэмминга, отлично подходит для дипломной работы в техникуме.

**Заявки**

*1****. Разработка программы «психотест: дом, дерево, человек». Идеи с чего можно начать:***

- Поиск в гугл «How to Label Images for Object Detection»

- <https://pro.arcgis.com/en/pro-app/latest/help/analysis/image-analyst/label-objects-for-deep-learning.htm>

- <https://towardsdatascience.com/how-to-label-images-for-object-detection-step-by-step-7ee317f98583>

- <https://www.esri.com/arcgis-blog/products/arcgis-pro/geoai/tips-for-labeling-images-for-object-detection-models/>

- Сменить вариант психологического теста на например, Myers-Briggs Personality Type Dataset, ссылка – <https://www.kaggle.com/datasets/datasnaek/mbti-type>

Вариант с GitHub <https://github.com/ianscottknight/Predicting-Myers-Briggs-Type-Indicator-with-Recurrent-Neural-Networks>

***2. Приложение на андроид для слабослышащих, переводящее аудио в текст с отправкой по почте и соцсетям.***

**1. Определите функциональные требования:**

Распознавание аудио: Используйте библиотеки для распознавания речи, такие как Google Speech-to-Text API или **Yandex Speech**.

Перевод в текст: Обработайте распознанный текст с помощью соответствующих алгоритмов или API.

Интеграция с электронной почтой: Реализуйте функциональность отправки текста по электронной почте через стандартные Android-функции или сторонние библиотеки.

Интеграция с социальными сетями: Предусмотрите возможность публикации текста в социальных сетях через соответствующие API.

**2. Разработайте пользовательский интерфейс:**

Создайте простой и интуитивно понятный интерфейс для удобства использования слабослышащими.

Разместите элементы управления, такие как кнопки записи, отправки текста и настройки, в удобном месте.

**3. Интегрируйте библиотеки распознавания речи:**

Подключите библиотеки для распознавания речи в ваш проект. Настройте их в соответствии с вашими потребностями.

**4. Добавьте функциональность перевода текста:**

Используйте соответствующие алгоритмы или API для перевода распознанного текста в нужный язык.

**5. Реализуйте отправку по почте и в социальные сети:**

Интегрируйте функциональность отправки текста через электронную почту, используя стандартные Android-инструменты или сторонние библиотеки.

Добавьте возможность публикации текста в социальных сетях, используя соответствующие API.

**6. Обеспечьте безопасность:**

Защитите данные пользователя, особенно если приложение требует доступ к личной информации.

**7. Проведите тестирование:**

Осуществите тестирование приложения на различных устройствах и разрешениях экрана.

Проверьте правильность распознавания речи и отправки текста.

**8. Опубликуйте приложение:**

Зарегистрируйте приложение в Google Play Store и следуйте указаниям по публикации.

**9. Поддерживайте и обновляйте:**

Отслеживайте обратную связь от пользователей и регулярно обновляйте приложение, улучшая его функциональность и исправляя возможные ошибки.

Обратите внимание, что для использования определенных сервисов, таких как Google Speech-to-Text API, может потребоваться регистрация и получение API-ключа. Также убедитесь, что ваше приложение соблюдает законы и нормативы в области конфиденциальности данных.

**Быстрый туториал с Хабра:** [**https://habr.com/ru/articles/681566/**](https://habr.com/ru/articles/681566/)

**Полный туториал часть 1:** [**https://thecode.media/stt-cloud/**](https://thecode.media/stt-cloud/)

**Полный туториал часть 2:** [**https://thecode.media/stt-python/**](https://thecode.media/stt-python/)

***3. Нейросеть для генерации рецептом по фото продуктов из холодильника***

Создание нейросети для генерации рецептов по фотографии продуктов из холодильника — это сложная задача, требующая не только знания в области машинного обучения, но и определенного опыта в обработке изображений и нейронных сетях. Вот общий план, как вы можете приступить к этой задаче:

**1. Подготовка данных:**

Соберите набор данных, содержащий изображения продуктов и соответствующие им рецепты. Обратите внимание, что это может потребовать тщательной разметки данных.

Подготовьте данные для обучения, включая их нормализацию и разделение на тренировочные и тестовые наборы.

**2. Выбор архитектуры нейронной сети**:

Выберите подходящую архитектуру нейронной сети. Это может быть рекуррентная нейросеть (RNN), сверточная нейросеть (CNN) или их комбинация.

Рассмотрите использование предобученных моделей, таких как VGG, Inception или ResNet, для извлечения признаков.

**3. Обучение модели:**

Настройте параметры обучения, такие как скорость обучения, количество эпох и т.д.

Используйте функции потерь, которые подходят для вашей задачи. Это может включать в себя использование как функции потерь для классификации, так и функции потерь для генерации текста (например, перекрестная энтропия или среднеквадратичная ошибка).

**4. Обработка изображений:**

Реализуйте механизм обработки изображений в вашей модели. Может потребоваться использование предварительных обученных моделей для определения объектов на фотографии.

**5. Генерация рецептов:**

Реализуйте механизм генерации текста, который может создавать рецепты на основе извлеченных признаков изображений.

**6. Тестирование и настройка:**

Оцените производительность модели на тестовом наборе данных. Убедитесь, что она создает реалистичные и полезные рецепты.

Внесите коррективы в архитектуру или параметры обучения при необходимости.

**7. Развитие и оптимизация:**

Используйте обратную связь от пользователей и дорабатывайте модель для улучшения качества генерируемых рецептов.

Оптимизируйте производительность модели, чтобы она могла работать эффективно на мобильных устройствах или в онлайн-сервисах.

**8. Внедрение:**

Интегрируйте вашу модель в приложение или веб-сервис для использования конечными пользователями.

Учтите, что создание такой нейросети требует не только программирования на Python с использованием библиотек машинного обучения, таких как TensorFlow или PyTorch, но и понимания основных концепций обработки изображений и обработки естественного языка.

**Готовый скрипт Recipe Generation using Recurrent Neural Network (RNN)**

In this experiment we will use character-based [Recurrent Neural Network](https://colab.research.google.com/corgiredirector?site=https%3A%2F%2Fen.wikipedia.org%2Fwiki%2FRecurrent_neural_network) (RNN) to generate cooking recipes. We will try to teach our RNN to generate recipe *name*, *ingredients* and *cooking instructions* for us.

I don't expect the RNN to do a strong connection between list of ingredients and cooking instructions but I do expect RNN to learn English grammar and punctuation in couple of hours and to generate some meaningful recipe names along with real food ingredients and cooking instructions.

For this experiment we will use [Tensorflow v2](https://colab.research.google.com/corgiredirector?site=https%3A%2F%2Fwww.tensorflow.org%2F" \t "_blank) with its [Keras API](https://colab.research.google.com/corgiredirector?site=https%3A%2F%2Fwww.tensorflow.org%2Fguide%2Fkeras" \t "_blank).

***⚠️ The recipes in this notebook are generated just for fun and for learning purposes. The recipes are not for actual cooking****!*



Ссылка на код: <https://colab.research.google.com/github/trekhleb/machine-learning-experiments/blob/master/experiments/recipe_generation_rnn/recipe_generation_rnn.ipynb#scrollTo=dwEfGi3mB2vM>

Он же в режиме браузера: <https://trekhleb.dev/machine-learning-experiments/#/experiments/RecipeGenerationRNN>

Его подробное описание: <https://www.kdnuggets.com/2020/07/generating-cooking-recipes-using-tensorflow.html>

**Генерация рецепта из изображения еды**

Are you ever in the situation where you have a mouthwatering picture of a delicious dish but no idea how to recreate it? Look no further! Our "Recipe Generation from Food Images" project harnesses the power of deep learning to provide you with a solution.

**Overview:**

This innovative application uses state-of-the-art deep learning techniques to analyze food images and generate comprehensive cooking recipes. From a simple snapshot of your favorite dish, our system can extract essential information including:

**Recipe Title**: A catchy and descriptive title for the dish.

**Ingredients**: A detailed list of all the ingredients needed.

**Instructions**: Step-by-step cooking instructions to help you recreate the dish flawlessly.

**Key Features:**

**Deep Learning Magic**: Our model employs cutting-edge computer vision algorithms to identify ingredients and cooking processes within food images.

**Natural Language Generation**: It utilizes advanced natural language processing to create coherent and easy-to-follow cooking instructions.

**User-Friendly Interface**: A user-friendly web or mobile interface for hassle-free recipe generation.

**Endless Culinary Exploration**: Discover new recipes and cuisines by simply snapping a picture of any dish.

**Ссылка на GitHub:** <https://github.com/navassherif98/Recipe-Generation-from-Food-Image/tree/main>