1 слайд

Здравствуйте уважаемые члены комиссии.

2 слайд

С развитием цифровых технологий и ростом интернет-активности существенно возросла потребность в защите web-ресурсов от автоматизированного взаимодействия. Одним из ключевых инструментов такой защиты являются системы CAPTCHA.

Современные системы CAPTCHA предлагают множество форматов защиты, но одновременно с этим появляются возможности для их автоматического распознаванию, в том числе с использованием методов машинного обучения и нейросетевых архитектур.

3 слайд

Целью данной работы является разработка и анализ комплексного подхода к автоматизации решения CAPTCHA в различных форматах с использованием современных нейросетевых инструментов и API для распознавания.

4 слайд

Для достижения поставленной цели были сформулированы следующие задачи, представленные на данном слайде.

5 слайд

Исторически распространенный тип CAPTCHA был впервые изобретен в 1997 году двумя группами, работающими параллельно. Эта форма CAPTCHA требует ввода последовательности букв или цифр из искаженного изображения.

Набравшая популярность технология reCAPTCHA, была приобретена Google в 2009 году. В дополнение к предотвращению мошенничества с ботами для пользователей, Google использовал технологию reCAPTCHA для оцифровки архивов The New York Times и книг из Google Books в 2011 году.

6 слайд

Данный формат CAPTCHA представлен на текущем слайде. На подобных CAPTCHA, зачастую использовались искажения, среди которых:

1. геометрические искажения;
2. перекрытие символов;
3. добавление шума;
4. нелинейные искажения.

7 слайд

На сегодняшний день CAPTCHA является важной мерой безопасности, так как предотвращает автоматические атаки. Современные системы CAPTCHA используют не только текст, но и аудио, изображения, поведенческие анализы и другие инновационные подходы, чтобы сделать тесты удобными для людей, но сложными для программ.

8 слайд

CAPTCHA в формате изображений, на сегодняшний день, широко используется для защиты ресурсов от автоматизированных ботов и может быть реализована несколькими способами. Наиболее распространены два варианта реализации, которые представлены на слайде:

1. цельное изображение, содержащее несколько объектов, частично размытых или искаженных, при этом изображение разбито на сетку 3×3 или 4×4.;
2. составное изображение, сформированное из 9 или 12 отдельных фрагментов (изображений), каждый из которых представляет собой независимое изображение – зачастую низкого качества, с наложением артефактов или шумов.

9 слайд

В ходе тестирования различных подходов для автоматизации решения CAPTCHA в различных форматах были выбраны наиболее подходящие для каждого конкретного формата. Подходы, которые использовались в данной работе представлены на слайде.

10 слайд

На данном слайде представлена блок-схема алгоритма распознавания голосового сообщения в аудио CAPTCHA. По умолчанию модель может работать только с аудиокодеками высокой четкости, а все аудиозаписи для CAPTCHA представлены в MP3. Перекодирование осуществляется с помощью библиотеки ffmpeg.

После этого, перекодированный файл загружается для передачи на вход модели для выделения голоса и преобразования его в текст. По завершению данной операции текстовое сообщение сохраняется для последующего использования, а исходный и перекодированный аудиофайлы удаляются. Блок-схема также представлена на раздаточном материале.

11 слайд

На данном слайде представлена блок-схема автоматизированного сценария с использованием фреймворка Selenium для прохождения CAPTCHA в аудиоформате с использованием алгоритма распознавания голосового сообещния, показанного на предыдущем слайде.

Согласно данному сценарию, сначала необходимо получить доступ непосредственно к заданию. Для этого осуществляется инициализация настроек браузера и переход на целевой сайт. Далее, в DOM-структуре web-страницы осуществляется поиск чекбокса «Я не робот» и эмулируется нажатие на него. После чего, в новом окне осуществляется поиск элемента для перехода непосредственно к аудиозаданию и нажатие на этот элемент.

После появление окна с аудиозаданием инициируется запрос на получение файла по извлеченной со страницы ссылке на него и сохранение файла локально. Затем, сохраненный файл подается в подпрограмму для распознавания голоса и возвращенный ею результат распознавания вставляется в текстовое поле. Для завершения решения эмулируется нажатие на кнопку подвтерждения ввода. Блок-схема также представлена на раздаточном материале.

12 слайд

Текстовые CAPTCHA на сегодняшний день уже не являются настолько же широкоиспользуемыми как CAPTCHA с изображениями, в связи с чем по-

лучение достаточного количества изображений для формирования датасета является трудоемкой задачей.

Качество используемого датасета оказывает существенное влияние на итоговую точность работы модели. Для эффективного обучения необходимо, чтобы набор данных соответствовал следующим требованиям:

1. достаточное количество изображений для каждого символа;
2. разнообразие данных;
3. переменная длина последовательностей символов.

Поскольку в открытом доступе отсутствует достаточное количество данных для формирования сбалансированного датасета, необходимо использовать другие способы для получения разнообразных примеров. Среди таких способов наиболее удобным и подходящим для данной задачи является генерация синтетических изображений с использованием специализированных библиотек. В качестве основного инструмента для решения данной задачи, зачастую, используется библиотека captcha на языке Python, обладающая необходимым функционалом для создания изображений CAPTCHA с заданными параметрами. Данная библиотека поддерживает генерацию изображений с пользовательскими шрифтами и различными эффектами искажений, что исключает необходимость привлечения дополнительных инструментов.

После создания изображений все они прошли этапы предобработки, направленные на улучшение качества данных и повышение эффективности обучения модели. Предобработка включала следующие этапы:

1. преобразование изображений в градации серого для уменьшения количества каналов и снижения вычислительной нагрузки;
2. бинаризация изображений с целью получения контрастного представления символов (белый текст на черном фоне);
3. удаление шумов и фона с использованием морфологических операций, в частности, дилатации.

Результат предобработки показан на данном слайде.

13 слайд

Для проведения экспериментов исходный набор данных, содержащий 100 000 изображений, был случайным образом перемешан и разделен на три подмножества: обучающее, тестовое и валидационное в соотношении

80:10:10. Обучающая выборка использовалась непосредственно для обучения модели, валидационная – для контроля качества процесса обучения на каждой эпохе, а тестовая – для окончательной оценки модели на данных, с которыми она ранее не сталкивалась.

В процессе многократного обучения были экспериментально определены оптимальное количество эпох и значения гиперпараметров, обеспечивающие эффективное снижение функции потерь до приемлемых значений. График изменения функции потерь представлен на слайде.

Для предотвращения переобучения использовался механизм ранней остановки, согласно которому обучение прекращалось при отсутствии уменьшения значения функции потерь на валидационной выборке в течение трех последовательных эпох. В данном эксперименте обучение завершилось на 18-й эпохе. На графике видно, что функция потерь стабилизировалась после 10 эпохе, поэтому 10 эпоха является балансом между точностью распознавания последовательностей и скоростью обучения модели.

Анализ графика сходимости функции потерь показывает наличие резкого увеличения ее значения на 9-й эпохе, что может быть обусловлено следующими факторами:

1. перемешивание данных перед каждой эпохой могло привести к образованию несбалансированной выборки, содержащей значительное число сложных примеров.
2. динамическое изменение скорости обучения, осуществляемое с помощью механизма регулирования скорости обучения, могло повлиять на изменение функции потерь.

14 слайд

Также была построена матрица ошибок, позволяющая проанализировать частоту и характер ошибок модели при классификации различных классов. Данная матрица приведена на слайде и на раздаточном материале. На данной матрице видно, что каждый класс распознается с высокой точностью (выше 88%).

15 слайд

Окончательная точность распознавания отдельных символов составила 0.9263.

После подбора оптимальных значений гиперпараметров модель была сохранена и протестирована на тестовой выборке. Точность распознавания последовательностей различной длины представлена в таблице.

16 слайд

Для обеспечения высокой точности в задаче автоматического решения графических CAPTCHA необходимо подготовить собственный набор данных, приближенный к реальным условиям использования. Наиболее эффективным методом является автоматизированный парсинг изображений CAPTCHA, представленных на web-сайтах, использующих визуальные CAPTCHA-решения, такие как Google reCAPTCHA v2.

Полученные изображения и метаданные (включая текст задания и параметры сетки) используются для формирования обучающего датасета, пригодного для дообучения модели YOLOv8 в задачах классификации и сегментации объектов.

После получения достаточного количества изображений для составления датасета необходимо провести их предварительную обработку и разметку.

Для создания меток используется инструмент CVAT (Computer Vision Annotation Tool) – многофункциональное веб-приложение с поддержкой аннотации объектов с помощью полигонов, прямоугольников и других форм. На данном слайде представлен пример разметки изображения с использованием данного инструмента.

Также, разметка позволяет учесть сразу несколько объектов разных классов на одном изображении, что особенно характерно для CAPTCHA, где в одной сетке могут одновременно находиться, например, автомобили и автобусы.

17 слайд

Кроме того, для корректного обучения модели YOLO требуется создать иерархическую структуру папок, в которой изображения и соответствующие метки будут разделены на тренировочную и валидационную выборки. На данном слайде представлен конфигурационный файл для обучаемой модели.

18 слайд

Обучение проводилось на 35 эпохах при размере изображений 640×640 пикселей и размере батча 8.

Была построена нормализованная матрица ошибок для определения точности предсказания необходимых классов на валидационной выборке, которая представлена на слайде и на раздаточном материале. По причине того, что обучающая выборка была несбалансированной не удалось достичь высоких показателей точности для классов, которые встречаются реже, но при дальнейшем увеличении датасета и его разнообразия можно значительно улучшить показатели точности для проблемных классов.

19 слайд

Использование предобученных весов позволило достичь стабильного снижения функции потерь с первых эпох, а встроенные механизмы аугментации способствовали улучшению обобщающей способности модели, что видно на текущих графиках. Данные графики также представлены на раздаточном материале.

20 слайд

На данном слайде представлена блок-схема автоматизированного сценария с использованием фреймворка Selenium для прохождения графической CAPTCHA. На первом этапе требуется инициализировать настройки браузера и получить доступ к целевому web-ресурсу. После чего, осуществляется поиск чекбокса «Я не робот» на странице для доступа к заданию.

Решение задачи будет осуществляться до тех пор, пока окно с заданием не исчезнет, поскольку данный формат CAPTCHA может требовать прохождения нескольких задач последовательно. С окна с заданием парсится текст задания, изображение и размерность сетки таблицы. После чего, осуществляется автоматический перевод текста задания на английский язык и сингуляризация для последующей обработки моделью. Поскольку в некоторых вариациях CAPTCHA часть изображения может обновляться после нажатия на ячейку необходимо проверять во время каждой итерации, что изображение не обновилось, в противном случае потребуется обрабатывать изображение заново.

Далее, полученное изображение передается в модель, которая создает список масок из которых выбираются только маски с целевым объектом. Для полученных масок определяются координаты ячеек, в которых они находятся, после чего ячейки с такими номерами прокликиваются.

После нажатия на ячейку проверяется, не обновилось ли там изображение и если это так, то создается запрос на получение нового изображения и полученное изображение заменяет собой соответствующий фрагмент на локальном изображении и обновленное изображение снова подается на вход модели. Таким образом, после того, как все необходимые ячейки будут выбраны и новых объектов не появится будет нажата кнопка подтверждения выбора и либо задача будет решена и окно закроется, либо появится новое задание и цикл действий будет снова повторен. Блок-схема также представлена на раздаточном материале.

21 слайд

В результате выполнения были решены задачи, представленные на слайде и цель работы была достигнута.

22 слайд

Кроме того, были определены дальнейшие перспективы развития исследования, которые представлены на данном слайде. Спасибо за внимание.