Актуальность

CAPTCHA давно является стандартным инструментом для защиты веб-ресурсов от спама, автоматизированных ботов и нежелательного извлечения данных. Большинство современных сайтов и веб-приложений используют данную технологию в различных сценариях, включая регистрацию пользователей, подтверждение действий на сайте и защиту от автоматизированных атак.

Автоматизированное распознавание текстовых CAPTCHA позволяет значительно снизить необходимость ручного тестирования веб-приложений, что, в свою очередь, повышает скорость и эффективность тестирования. Кроме того, подобные методы могут использоваться для анализа надёжности внедрённых CAPTCHA, выявления их слабых мест и повышения безопасности веб-приложений, например, за счёт комбинирования нескольких методов защиты.

Современная реализация текстовых CAPTCHA

Современные текстовые CAPTCHA обычно состоят из букв и цифр. Зачастую используются символы латинского алфавита (как прописные, так и строчные) и цифры от 0 до 9. Но обычно реализации исключают символы, которые могут быть легко перепутаны, например, буквы «O» и цифру «0», буквы «I» и «l» и тому подобное. Рекомендуемый набор символов в генераторах на некоторых CRM платформах выглядит следующим образом: ABCDEFGHJKLMNPQRSTWXYZ23456789.

Длина последовательности символов в CAPTCHA обычно составляет от 4 до 8 символов, что обеспечивает баланс между удобством для пользователя и безопасностью, однако конкретная длина может варьироваться в зависимости от требований системы безопасности.

Для усложнения автоматического распознавания текстовые CAPTCHA подвергаются различным искажениям:

1. геометрические искажения: символы могут быть искажены, повернуты или наклонены, что затрудняет их распознавание автоматическими системами;
2. перекрытие символов: символы могут быть расположены близко друг к другу или даже перекрываться, что усложняет их сегментацию и последующее распознавание;

3. добавление шума: на изображение могут быть добавлены различные шумы, такие как линии, точки или круги, чтобы затруднить распознавание символов;
4. сложные фоны: использование фонов с различными цветами или узорами, что делает выделение символов более сложным;
5. нелинейные искажения: применение нелинейных трансформаций к символам, что делает их форму менее предсказуемой для автоматических систем распознавания.

Эти методы направлены на повышение сложности автоматического распознавания CAPTCHA, сохраняя при этом относительную легкость распознавания для человека.

Для распознавания текста с переменной длиной последовательности в задачах CAPTCHA наиболее часто применяются следующие архитектуры нейронных сетей:

1. оптическое распознавание символов (OCR);
2. рекуррентные сверточные нейронные сети (CRNN);
3. архитектуры последовательного обучения (Seq-to-Seq).

Для обучения моделей был сформирован датасет из 100 000 изображений CAPTCHA, содержащих случайные последовательности символов длиной от 4 до 7. Такой объем данных позволяет добиться высокой обобщающей способности модели и снизить вероятность переобучения.

Оптическое распознавание символов (OCR Tesseract)

Изначально предполагалась реализация модели с использованием OCR, поскольку такие системы изначально разрабатывались для задач оптического распознавания текста. В качестве конкретной модели был выбран Tesseract.

Tesseract является одной из наиболее популярных систем OCR с открытым исходным кодом. Tesseract поддерживает более 100 языков, включая сложные письменности. В версии 4.0 в модель была интегрирована нейронная сеть на основе долговременной краткосрочной памяти (LSTM), что позволило существенно повысить точность распознавания, особенно при обработке сложных шрифтов и рукописного текста.

Для решения поставленной задачи предполагалось использовать предобученную модель Tesseract и дообучить её на специализированном датасете, содержащем изображения CAPTCHA с характерными искажениями. Однако в ходе экспериментов было установлено, что точность распознавания

последовательностей символов целиком составляла 0%, а точность для отдельных символов оказалась крайне низкой. Это связано с тем, что архитектура Tesseract недостаточно устойчива к искажениям, характерным для CAPTCHA, таким как деформация символов, наложение шумов и изменение углов наклона.

Таким образом, было принято решение отказаться от использования Tesseract в пользу более адаптированных к данной задаче моделей, таких как сверточные рекуррентные нейронные сети (CRNN) или модели последовательного обучения (Seq-to-Seq), обладающие высокой устойчивостью к вариативности и искажениям, характерным для CAPTCHA.

Рекуррентные сверточные нейронные сети (CRNN)

Сверточно-рекуррентные нейронные сети (CRNN) представляют собой гибридную архитектуру, сочетающую в себе возможности сверточных нейронных сетей (CNN) и рекуррентных нейронных сетей (RNN). Данный подход используется в задачах, связанных с обработкой последовательных данных, таких как распознавание текста, речь и видео.

Основное преимущество CRNN заключается в способности CNN-части извлекать пространственные признаки из изображений, тогда как RNN-часть позволяет учитывать временные зависимости между последовательными фрагментами данных.

Разработанная модель CRNN для распознавания CAPTCHA включает в себя три ключевых блока:

1. сверточный блок (CNN): предназначен для выделения признаков из изображений CAPTCHA. Включает в себя три последовательных сверточных слоя, а также методы нормализации и уменьшения размерности признакового пространства;
2. рекуррентный блок (RNN): использует двунаправленные слои GRU, позволяющие модели учитывать зависимость между последовательными символами в CAPTCHA;
3. выходной слой: полносвязный, который выполняет классификацию каждого символа в последовательности.

В данной архитектуре применяются слои Dropout для регуляризации, также используется l2-регуляризация, BatchNormalization для ускорения обучения и повышения устойчивости модели, а также функция softmax для предсказания классов символов.

После обучения данной модели результаты оказались превосходящими показатели OCR, однако все же не достигли удовлетворительного уровня. В частности, точность распознавания всей последовательности символов не превышала 10%, тогда как точность классификации отдельных символов составляла около 70%.

Архитектура последовательного обучения (Seq-to-Seq)

Модели последовательного преобразования (Seq-to-Seq) широко применяются для задач, связанных с обработкой последовательностей переменной длины. Они используются в таких областях, как машинный перевод, распознавание речи и анализ текстов. Данные модели основаны на архитектуре энкодера-декодера, где первый модуль кодирует входную последовательность в скрытое представление, а второй декодирует его в выходную последовательность.

Одним из ключевых элементов Seq2Seq является механизм внимания, который позволяет декодеру динамически фокусироваться на различных частях входной последовательности при генерации выходных символов. Этот подход особенно полезен для распознавания CAPTCHA, так как символы в изображениях могут иметь разную ориентацию и степень искажения.

Кодировщик, в данной модели принимает входное изображение CAPTCHA и преобразует его в компактное представление. Архитектура кодировщика включает:

1. четыре сверточных блока, слои пакетной нормализации и слои подвыборки для понижения размерности входных данных;
2. глобальный усредненный слой для получения векторного представления изображения;
3. полносвязный слой для финального представления скрытого состояния;
4. рекуррентный слой для кодирования последовательности, возвращающий последнее скрытое состояние кодировщика.

Декодировщик выполняет пошаговую генерацию выходной последовательности, используя скрытое состояние кодировщика. В архитектуру декодировщика входят:

1. входной слой для последовательности токенов;
2. слой вложения, который преобразует входные токены в векторные представления;

3. рекуррентный слой, обрабатывающий последовательность с учетом скрытого состояния кодировщика;
4. механизм внимания, который позволяет декодеру учитывать релевантные части входного изображения;
5. полносвязный слой с функцией активации softmax для предсказания вероятностей символов.

На начальных этапах экспериментов предложенная Seq-to-Seq-модель показала наилучшие результаты среди рассмотренных вариантов. В отличие от OCR- и CRNN-моделей, данная архитектура смогла достичь более высокой точности распознавания последовательностей символов, что обусловлено применением механизма внимания. Дальнейшая работа с моделью была сосредоточена на её оптимизации и улучшении параметров обучения.

Тестирование выбранной модели нейронной сети

Для проведения экспериментов исходный набор данных, содержащий 100 000 изображений, был случайным образом перемешан и разделён 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 качестве основных метрик качества модели использовались функция потерь (loss) и точность (accuracy), рассчитываемая для каждого символа последовательности.

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

Для предотвращения переобучения использовался механизм ранней остановки, согласно которому обучение прекращалось при отсутствии уменьшения значения функции потерь на валидационной выборке в течение трёх 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 её 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 (learning rate scheduler), могло повлиять на изменение функции потерь.

Окончательная точность распознавания отдельных символов составила 0.9263.

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

Также была построена матрица ошибок, позволяющая проанализировать частоту и характер ошибок модели при классификации различных классов.

Анализ полученных результатов показывает, что точность распознавания последовательностей значительной длины остаётся относительно низкой. Это можно объяснить высокой зависимостью модели Seq-to-Seq от объёма обучающих данных: для эффективного обобщения признаков, извлекаемых из изображений, требуется значительное количество примеров. Следовательно, увеличение размера обучающего набора данных потенциально может способствовать повышению точности модели, однако это также накладывает дополнительные требования к вычислительным ресурсам, необходимым для её обучения.