

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

**«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИМЕНИ Н. Г. ЧЕРНЫШЕВСКОГО»**

Кафедра математической кибернетики и компьютерных наук

**РАСПОЗНАВАНИЕ АНОМАЛЬНЫХ СОБЫТИЙ НА ВИДЕО С
ПОМОЩЬЮ ПОЛНОСВЯЗНОЙ НЕЙРОННОЙ СЕТИ**

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

Студентки 4 курса 451 группы
направления 09.03.04 — Программная инженерия
факультета КНиИТ
Крулевой Маргариты Михайловны

Научный руководитель

к. ф.-м. н., доцент

А. С. Иванов

Заведующий кафедрой

к. ф.-м. н., доцент

С. В. Миронов

Саратов 2022

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	3
1 Искусственные нейронные сети	4
1.1 Нейроны	4
1.2 Функция активации	4
1.3 Архитектуры нейронных сетей	4
1.4 Обучение нейронных сетей	4
1.5 Многоэкземплярное обучение	5
1.6 Функция потерь при ранжировании	5
1.7 Извлечение признаков	5
1.8 Нормализация	5
1.9 Переобучение и методы его предотвращения	6
1.10 Оптимизаторы	6
1.11 Валидация	7
2 Программная реализация	8
2.1 Подготовка данных	8
2.1.1 Используемые язык и библиотеки для реализации задачи ..	8
2.1.2 Набор данных	8
2.1.3 Извлечение признаков	8
2.1.4 Загрузка извлеченных признаков	8
2.2 Построение, обучение и тестирование модели	8
2.2.1 Построение полносвязной сети	8
2.2.2 Обучение модели	9
2.2.3 Функция потери при ранжировании	9
2.2.4 Построение ROC-кривой	9
2.3 Пользовательский интерфейс	9
ЗАКЛЮЧЕНИЕ	10
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	11

ВВЕДЕНИЕ

Актуальность темы. На сегодняшний день камеры видеонаблюдения используются во многих общественных местах для повышения безопасности. Одной из важнейших задач видеонаблюдения является обнаружение аномальных событий, таких как дорожно-транспортные происшествия, пожары или преступные действия.

Таким образом, чтобы снизить затраты труда и времени на мониторинг видеозаписей, разработка интеллектуальных алгоритмов компьютерного зрения для автоматического определения временного окна возникновения аномалии на видео становится необходимостью.

Цель бакалаврской работы — разработка программного обеспечения для обнаружения на видео ситуаций, отклоняющихся от нормы.

Поставленная цель определила **следующие задачи**:

1. Рассмотреть сверточные, полносвязные и многослойные нейронные сети.
2. Исследовать процесс обучения нейронных сетей и подход многоэкземплярного обучения.
3. Рассмотреть проблему переобучения ИНС и методы ее предотвращения.
4. Подобрать набор данных для обучения.
5. Извлечь характеристики из набора с помощью предобученной модели Convolution 3D.
6. Построить модель полносвязной сети.
7. Обучить модель на обработанных данных.
8. Разработать пользовательский интерфейс для отображения результатов работы модели.

Структура и объём работы. Бакалаврская работа состоит из введения, двух разделов, заключения, списка использованных источников и двух приложений. Общий объем работы — 42 страницы, включая 8 рисунков, цифровой носитель в качестве приложения, список использованных источников информации — 20 наименований.

1 Искусственные нейронные сети

1.1 Нейроны

На вход нейрон принимает n -мерный вектор признакового описания и свободный член, имеющий постоянное значение с весом. На выход нейрон выдает значение функции активации от взвешенной суммы входных значений. Существует три вида слоев: видимые, скрытые и выходные.

1.2 Функция активации

В качестве аргумента функция активации принимает взвешенную сумму входных значений нейрона, а ее значением является выход нейрона.

Линейная функция прямо пропорциональна взвешенной сумме нейронов и определена в диапазоне $(-\infty; +\infty)$. Функция ReLU является нелинейной, может применяться в глубоких нейронных сетях, определена в диапазоне $[0; \infty)$. Значения сигмоиды не бинарны и находятся в диапазоне $[0; 1]$. Ее значения стремятся к одной из сторон кривой.

1.3 Архитектуры нейронных сетей

В полносвязной нейронной сети прямого распространения каждый нейрон связан со всеми нейронами, находящимися в следующем слое, и все связи направлены строго от входных нейронов к выходным, не образуя обратных связей.

В многослойной нейросети имеется один или более скрытых слоев, высокая связность, каждый нейрон сети имеет нелинейную всюду дифференцируемую функцию активации.

СНС используется для глубокого обучения и содержит входной слой, чередующуюся последовательность слоев свертки и субдискретизации, выходной слой. Свертки могут применяться в N измерениях. Для работы с видео применяют трехмерные свертки [1].

1.4 Обучение нейронных сетей

Выделяют четыре вида обучения: контролируемое, неконтролируемое, полуконтролируемое, с подкреплением. При контролируемом обучении имеется выходная метка в наборах данных. При неконтролируемом обучении используются данные без входной метки, алгоритм пытается самостоятельно найти

корреляции. Для полуконтролируемого обучения используется набор из размеченных и неразмеченных данных. В обучении с подкреплением применяется система поощрений.

1.5 Многоэкземплярное обучение

При многоэкземплярном обучении обучающие экземпляры организованы в наборы, называемые пакетами, и для всего пакета предусмотрена метка. Все отрицательные пакеты содержат только отрицательные экземпляры, а положительные пакеты содержат по крайней мере один положительный экземпляр [2]. МЛ подходит для решения таких задач, как классификация, кластеризация, регрессия, ранжирование.

Задача кластеризации состоит в поиске кластеров или структуры среди набора немаркированных пакетов. Задача регрессии МЛ заключается в присвоении реального значения пакету (или экземпляру) вместо метки класса. Цель задачи ранжирования состоит в том, чтобы сравнить величину оценок для выполнения сортировки [3].

1.6 Функция потерь при ранжировании

Функция потерь используется для расчета ошибки между реальными и полученными ответами. Поскольку пакет представляет собой последовательность сегментов, оценка должна плавно меняться между сегментами. Таким образом, обеспечивается временная гладкость между оценками аномалий смежных во времени сегментов, сводя к минимуму разницу оценок для смежных сегментов [4]. При потере при ранжировании ошибка распространяется обратно от максимально оцененных сегментов как в положительных, так и в отрицательных пакетах.

1.7 Извлечение признаков

Извлечение признаков — это процесс уменьшения размерности, при котором исходный набор необработанных данных разделяется и сводится к более управляемым группам. Извлечение характеристик может выполняться вручную или автоматически.

1.8 Нормализация

Нормализация — это процесс перевода данных в диапазон $[0; 1]$ или $[-1; 1]$. Метод нормализации помогает улучшить производительность и точность моде-

ли.

1.9 Переобучение и методы его предотвращения

При переобучении модель дает хорошие результаты на обучающем наборе, но плохо работает на тестовом. Существует множество методов для предотвращения переобучения [1]:

1. Удаляя определенные слои или уменьшая количество нейронов, сеть становится менее подверженной переобучению, поскольку нейроны, способствующие переобучению, удаляются или деактивируются.
2. Увеличивая размер обучающего набора данных, сеть не может переобучить все входные образцы и вынуждена обобщать.
3. Регуляризация штрафует за большие веса сети и заставляет алгоритм оптимизации уменьшать большие значения, это приводит к стабильности сети и обеспечивает хорошую производительность.
4. Регуляризация L_1 сводит весовые значения менее важных признаков к нулю и исключает их из дальнейших вычислений.
5. Регуляризация L_2 применяется для решения проблемы мультиколлинеарности за счет ограничения коэффициента и сохранения всех переменных.
6. Отсевы деактивируют определенное количество нейронов на тренировочных данных во время обучения с вероятностью p . Удаленный нейрон возвращает 0 при любых входных данных или параметрах и не принимает участие в процессе обучения.
7. Ранняя остановка дает представление о том, сколько итераций можно запустить, прежде чем сеть начнет переобучаться. Обучение модели прекращается, когда производительность модели перестает улучшаться в наборе данных проверки.

1.10 Оптимизаторы

Оптимизаторы — это методы, используемые для изменения атрибутов нейронной сети, таких как веса и скорость обучения с целью снижения потерь. Оптимизация производится для сравнения результатов на каждой итерации, изменяя гиперпараметры на каждом шаге, пока не достигнуты оптимальные результаты.

Стохастический Градиентный Спуск (SGD) пытается чаще обновлять параметры модели. При этом параметры модели изменяются после вычисления

потерь на каждом обучающем примере. Адаград изменяет скорость обучения для каждого параметра и на каждом временном шаге, работает на производной функции ошибки. Адам основан на адаптивной оценке моментов первого и второго порядка, вычисляет экспоненциальное скользящее среднее градиентов и квадратных градиентов [5].

1.11 Валидация

Валидация предназначена для проверки, что модель работает так, как она была задумана, и что она решает проблему, для решения которой она была разработана.

2 Программная реализация

2.1 Подготовка данных

2.1.1 Используемые язык и библиотеки для реализации задачи

Программное обеспечение разработано на языке Python. Для создания основного функционала использована библиотека PyTorch.

2.1.2 Набор данных

В качестве обучающего набора данных был выбран датасет UCF-Crime [6]. В качестве обучающего набора данных был выбран датасет для обнаружения аномалий UCF-Crime. Он состоит из 1900 длинных и необрезанных реальных записей с камер видеонаблюдения, содержащих 13 аномалий, таких как драки, дорожно-транспортные происшествия, взрывы, грабежи и т. д., а также нормальные события. Общая длительность всех видео — 128 часов [6].

Выбранный набор данных слабо размечен, то есть известен только факт наличия или отсутствия аномалии в видео. Поэтому в работе используется метод многоэкземплярного обучения.

2.1.3 Извлечение признаков

Для извлечения признаков из набора данных UCF-Crime используется предобученная модель C3D.

2.1.4 Загрузка извлеченных признаков

Полученные признаки используются для формирования партии из случайным образом выбранных 30 аномальных и 30 нормальных пакетов.

2.2 Построение, обучение и тестирование модели

2.2.1 Построение полносвязной сети

Полученные характеристики вводятся в трехслойную полносвязную сеть [7]. Первый полносвязный слой имеет 512 узлов, за которыми следуют слои с 32 и 1 узлами. Между полносвязными слоями есть слои отсева 60% нейронов. Также используются активация ReLU и активация Sigmoid для первого и последнего полносвязных слоев соответственно.

2.2.2 Обучение модели

Основное предположение предлагаемого подхода заключается в том, что при наличии большого количества положительных и отрицательных видео с метками на уровне видео сеть может автоматически научиться предсказывать время аномалии в видео. Обучение проходило в течении 20 эпох по 20000 итераций. Был использован оптимизатор Adagrad с начальной скоростью обучения 0,01 и функция потерь при ранжировании.

2.2.3 Функция потерь при ранжировании

Функция определяет насколько хорошо работает модель, сравнив то, что модель прогнозирует, с фактическим значением. Если y_pred очень далеко от y_true , значение потерь будет очень высоким. Однако, если оба значения почти одинаковы, значение потерь будет очень низким [8].

2.2.4 Построение ROC-кривой

Результат работы алгоритма на фиксированной тестовой выборке визуализируется с помощью ROC-кривой, а качество оценивается как площадь под этой кривой — AUC [5]. Для построения кривой для каждого сегмента сравниваются реальные значения и предсказанные моделью. Для построения кривой используются функции `roc_curve`, `auc` из библиотеки `sklearn.metrics`. В результате значение AUC составило 0.68.

2.3 Пользовательский интерфейс

Медиапроигрыватель написан с помощью библиотеки PyQt5. На него добавлены кнопки загрузки и включения видео, слайдер для перемотки видео, область для отображения статуса и процента выполнения загрузки, а также область для построения графика [9]. при открытии файла происходит извлечение характеристик из загруженного видео, а затем выполнение предсказания.

Одновременно с показом видео отрисовывается график аномальных событий. По оси x — время видео, а по оси y — предсказание. Чем график по оси y ближе к единице, тем больше вероятность, что в этот момент на видео происходило аномальное событие.

ЗАКЛЮЧЕНИЕ

В результате дипломной работы были изучены теоретические основы ИНС, подробно рассмотрен процесс обучения нейронных сетей и подход многоэкземплярного обучения, исследована проблема переобучения и методы ее предотвращения. На языке Python разработано приложение для распознавания аномальных событий на видео, позволяющее без затрат большого количества времени обнаружить временной интервал возникновения нестандартной ситуации.

В процессе был использован набор данных UCF-Crime. С помощью предобученной модели C3D из него были извлечены характеристики для обучения модели построенной трехслойной полносвязной сети. А также с применением библиотеки PyQt5 разработан интерфейс, используя который пользователь может загрузить любое видео и получить график с отображением предсказанных меток для каждого момента видео.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- 1 *Форсайт, Д.* Компьютерное зрение. Современный подход / Д. Форсайт, Д. Понс. — М.: Вильямс, 2004.
- 2 *Carbonneau, M.-A.* Multiple instance learning: A survey of problem characteristics and applications / M.-A. Carbonneau, V. Cheplygina, E. Granger, G. Gagnon // *Pattern Recognition*. — 12 2016. — P. 77.
- 3 *Bergeron, C.* Multiple instance ranking. — 01 2008. — Pp. 48–55.
- 4 *Menon, A. K.* Predicting accurate probabilities with a ranking loss // ICCV. — 2019.
- 5 *Шолле, Ф.* Глубокое обучение на Python / Ф. Шолле. — Питер, 2018. — С. 400.
- 6 *Sultani, W.* Real-world anomaly detection in surveillance videos // The IEEE Conference on Computer Vision and Pattern Recognition (CVPR). — June 2018.
- 7 *Rosebrock, A.* Deep Learning for Computer Vision with Python: Starter Bundle / A. Rosebrock. — 1 edition. — PyImageSearch, 2017.
- 8 *Zhang, B.* Dynamic feature regularized loss for weakly supervised semantic segmentation / B. Zhang, J. Xiao, Y. Zhao. — 2021. — P. 8.
- 9 *Fitzpatrick, M.* Create GUI Applications with Python and Qt5 (PyQt5 Edition): The hands-on guide to making apps with Python / M. Fitzpatrick. — 4 edition. — 2020. — P. 689.