МИНОБРНАУКИ РОССИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ Н. Г. ЧЕРНЫШЕВСКОГО»
Кафедра дискретной математики и информационных технологий

МЕТОДЫ ГЛУБОКОГО ОБУЧЕНИЯ ДЛЯ АНАЛИЗА ТЕКСТОВ

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

студента 4 курса 421 группы
направления 09.03.01 — Информатика и вычислительная техника
факультета КНиИТ
Ефремова Данилы Алексеевича

Научный руководитель
к. ф.-м. н., д. э. н., профессор _____ Л. В. Кальянов

Заведующий кафедрой
доцент, к. ф.-м. н. _____ Л. Б. Тяпаев

Саратов 2023

ВВЕДЕНИЕ

Глубокое обучение нейронных сетей в настоящее время является одним из наиболее актуальных и перспективных направлений в области искусственного интеллекта. Эта технология стала основой для ряда инновационных приложений в различных сферах, таких как компьютерное зрение, обработка естественного языка, голосовые помощники, медицинская диагностика и другие.

Цель данной дипломной работы заключается в изучении принципов и методов глубокого обучения нейронных сетей, а также в исследовании их применения в различных областях. В процессе исследования будет рассмотрен ряд важных аспектов, таких как архитектуры нейронных сетей, методы обучения, проблемы оптимизации, практические применения и тенденции развития данной области.

Актуальность выбранной темы обусловлена стремительным развитием технологий глубокого обучения и их потенциалом для решения сложных задач, которые ранее считались невозможными для автоматизации. Поэтому изучение данной темы имеет важное значение как для науки, так и для практики, и может способствовать созданию новых инновационных продуктов и сервисов, улучшающих качество жизни людей.

Для достижения поставленной цели были поставлены следующие задачи:

- Разобрать методы глубокого обучения
- Разобрать особенности методов анализа текстов.
- Разобрать средства Python для реализации методов глубокого обучения
- Реализовать пример анализа текстов (на примере дата сета из открытых источников)

КРАТКОЕ СОДЕРЖАНИЕ РАБОТЫ

В первой главе представлены теоретические сведения об устройстве нейронной сети. Перечислены основные виды нейронных сетей,виды их обучения. Акцент сделан на глубокое обучение, которое и является основной темой данной работы.

Существует стандартное представление нейросети, где выходной сигнал одного нейрона передается на вход другому нейрону. Сеть может состоять из большого количества нейронов, объединенных в слои. Входной слой получает сигналы от внешнего мира. Скрытый слой получает сигналы от входного слоя и выдает сигнал на выходной слой, который в свою очередь выдает сигнал во внешнюю среду. Слой называется скрытым, т.к. того что в нем происходит не видно из внешней среды.

Существует 2 основных типа нейронных сетей. Первый тип - это сеть с прямым распространением сигнала. В такой сети запрещены циклы. Сигнал, который поступает на входной слой, передаётся на скрытый слой, после чего передается на выходной слой и во внешнюю среду.

Другой тип сетей - рекуррентные сети. В рекуррентных сетях возможны циклы, что значит, что сигнал от одного нейрона может поступать к этому же нейроны, к другим нейронам того же слоя или даже нейронам предыдущего слоя.

Нейронные сети могут включать в себя несколько скрытых слоев. Сети, у которых больше одного скрытого слоя называются глубокими нейронными сетями.

Теперь, немного о том, что такое обучение нейронных сетей. Обучение нейронной сети - это подбор весов, которые соответствуют входам таким образом, чтобы сеть решала поставленную задачу.

Существуют 3 подхода к обучению.

Обучение с учителем, обучение без учителя и обучение с подкреплением.

В нейронных сетях чаще всего применяются обучение с учителем. В этом случае у нас должен быть набор данных, то есть сигналов которые подаются на вход нейронной сети для которых заранее правильный ответ другой подход

В случае обучения без учителя у нас есть данные, но правильный ответ

для них заранее неизвестен. Обучение без учителя основывается на выявлении структурных различий в данных, например, мы можем проводить анализ временных рядов, например, фотографии столов очень сильно отличаются от фотографий машин. Нейронная сеть способна самостоятельно выявить структурные различия в этих фотографиях и на основании этих структурных различий отнести фотографию либо к столам, либо к машинам. При это нейронная сеть не будет знать что объект который показан на фотографии это стол или машина, она делает выводы только на основе различий в структуре.

Третий вид обучения обучение с подкреплением. В этом случае у нас нет заранее подготовленных наборов данных. Наша сеть действует как агент во внешней среде и получает сигналы от внешней среды правильно выполняются или неправильно. Для обучения нейронных сетей чаще всего используются обучение с учителем .

Хотя обучение нейронных сетей - это достаточно сложная задача, в то же время она является типовой. Существует большое количество библиотек которые позволяют обучать нейронные сети, поэтому существует возможность достаточно быстро обучать нейронные сети.

В данной работе использовался язык Python. Также использовались библиотеки TensorFlow, Theano и Keras. Библиотека Keras сама построит необходимую сеть и для проведения вычислений будет сама вызывать высокоэффективные методы из библиотек TensorFlow и Theano.

Во второй главе реализовано глубокое обучение нейронных сетей на готовом датасете.

Примером глубокого обучения в данной работе будет являться классификация новостей с использованием AG's News Topic Classification Dataset.

Будут используется три архитектуры нейронных сетей: Одномерная сверточная нейросеть Рекуррентная нейросеть LSTM Рекуррентная нейросеть GRU

Первый столбец является числом класса текста. Второй столбец является заголовком. Третий - текстом самой новости. В отдельном файле указаны классы новостей. Используется всего 4 класса новостей. Каждую новость можно отнести только к одному классу.

Их структура аналогична обучающим данным. В первом столбце класс

новости, во втором - заголовок, в третьем - текст самой новости.

В данной работе проведено обучение трёх нейросетей. Для обучения сетей будет использоваться функция ошибки "cathegorical crossentrophy". Категориальная означает, что имеется несколько классов, в данном случае - 4. На выходе 4 нейрона.

Подключаем необходимые модули TensorFlow и Keras.

Задаем параметры обучения. Ограничиваем максимальное количество слов. Так как новости довольно короткие - ограничиваем максимальною длину новости и указываем количество доступных классов.

Загружаем данные для обучения.

Загружаем данные для тестирования.

Также дополнительно загружаем имена классов.

Проверяем загруженные файлы. На скриншоте видны файлы train.csv и test.csv для обучения и тестирования, соответственно, и файл classes.txt с названиями классов.

Проверяем содержание файла classes.txt.

Проверяем количество данных для обучения и тестирования.

Также проверяем соответствие локальных данных с загруженных файлов датасета и данных с файлов загруженных на виртуальную машину Google. Проверяем файл для обучения и для тестирования.

Как видно, содержание файлов соответствует изначальному.

Получаем количество - 120000 новостей для обучения, 7600 для тестирования. Далее загружаем данные в память для подготовки к обучению нейросетей.

Для этого используем библиотеку Pandas. Вручную вписываем заголовки. Первый заголовок будет называться class, второй - title,третий - text.

Проверим загруженные файлы в датафрейме Pandas.

Первый столбец class соответствует номеру класса, второй столбец title - заголовку статьи, третий столбец text - самой статье. Далее нужно выбрать данные для обучения и тестирования. На первом этапе будем использовать данные для обучения из класса text.

Для правильных ответов выбираем столбец train.class, вычитаем единицу, для того чтобы номера классов начинались с нуля, а не с единицы, и используем функцию to-cathegorical, чтобы номер класса представить к фор-

мату one-hot-encoding.

Получаются векторы из 4 значений, каждое из которых равно нулю, кроме вектора, соответсвующего номеру класса. Далее требуется преобразовать текст в числовое представление. Для этого еще раз проверим исходный текст.

Чтобы преобразовать текст в числовое представление будем использовать токенизатор Keras.

Обучаем токенезатор Keras на наших новостях. Здесь используется только текст без заголовков.

После обучения просмотрим полученный словарь токенезатора.

Наиболее часто встречаются артикли. Токенизатор будет заменять эти слова на соответствующие им числа.

Теперь выполним замену текстового представления на числовое.

Текстовое представление новости находится сверху, а числовое аналогично находится снизу. Слова всех новостей заменились на числа из словаря, которые построил токенизатор. Данные на вход нейронной сети должны быть одинаковой длинны, поэтому ограничиваем длину в 30 слов. После преобразования первые пять новостей будут иметь следующий вид.

Так как в примере первая и последняя новость короткие, они были дополнены нулями. Теперь когда данные готовы можно приступить к созданию нейронной сети. Первой создана сверточная нейронная сеть.

Запускаем обучение нейронной сети. X-train - это набор данных новостей, y-train - классы с правильными ответами, которые представлены в формате one-hot-encoding. Размер минивыборки - 128 элементов. 10 процентов данных будет использоваться для проверочного набора данных.

Согласно результату, лучшая доля правильных ответов находится на 1 эпохе. Построим график обучения.

Происходит переобучение. Доля правильных ответов на обучающем наборе данных показана синим увеличивается, а на проверочном наборе данных снижается.

Перейдем к следующей сети LSTM. В ней также используются четыре нейронные функции, Softmax на выходе и функцией ошибки является категориальная перекрестная энтропия.

Выводим информацию о сети.

Создаем callback для сохранения нейронной сети на каждой эпохе, если качество работы на проверочном наборе данных улучшилось. Сеть сохраняется в файл best-model.h5.

Производим обучение сети LSTM.

Согласно результату, лучшая доля правильных ответов находится на 2 эпохе. Эта сеть сохранена в файл best-model.h5. Построим график обучения.

Доля ответов на проверочном наборе данных сначала увеличивается, потом начинает снижаться, что так же как и в первом примере, говорит о переобучении.

Третий вариант нашей сети - сеть GRU. Главное отличие заключается в том, что на рекуррентном слое вместо ячеек LSTM ячейки GRU в количестве 16 единиц. (Слайд)

Выводим информацию о сети.

Создаем callback для сохранения нейронной сети на каждой эпохе, если качество работы на проверочном наборе данных улучшилось. Сеть сохраняется в файл best-model-gru.h5.

Производим обучение сети GRU.

Здесь, так же как и с сетью LSTM, лучший вариант сети получен на второй эпохе. Доля правильных ответов на проверочном наборе данных составляет 89,19Построим график обучения.

На графике обучения ситуация аналогичная графику сети LSTM. Теперь оценим качество работы всех сетей на тестовом наборе данных(рис.52). Для этого необходимо выполнить с тестовым набором данных, который находится в файле test.csv те же самые операции, которые были выполнены с обучающим набором данных.

Сначала загрузим его и создадим data frame с тремя столбцами: класс - номер класса, title - название новости и текст новости.

Далее используем токенизатор для того, чтобы преобразовать текстовое представление в числовое. При этом требуется использовать тот же самый токенизатор который обучен на наборе данных train, а не обучать токенизатор заново на наборе данных для тестирования. Ограничиваем длину новостей в 30 символов и смотрим как выглядят данные.

Из номера класса вычитаем единицу. А дальше с помощью функции to-cathegorical преобразуем их в представлении по категориям или one hot

encoding.

У нас 4 класса и 4 нейрона на выходе. Все равны нулю кроме одного, который соответствует номеру нужного нам класса. Теперь оценим качество работы всех вариантов сети. Сначала загружаем лучшие веса сверточной нейронной сети.

Оцениваем качество её работы на тестовом наборе данных на 88.9 процента. То же самое проделываем для сети LSTM.

Оцениваем качество её работы на тестовом наборе данных на 89.15 процента. И в конце аналогичные действия для сети GRU.

Качество её работы 89.56 процента. Исходя из этих данных лучший результат показала сеть GRU.

ЗАКЛЮЧЕНИЕ

В ходе выполнения данной дипломной работы были рассмотрены основные принципы и методы глубокого обучения нейронных сетей, а также их применение в различных областях. Исследование позволило выявить значительные достижения и перспективы развития данной технологии, а также выявить вызовы и проблемы, требующие дальнейших исследований.

Были выполнены следующие поставленные задачи:

— Разобраны методы глубокого обучения
— Разобраны особенности методов анализа текстов.
— Разобраны средства Python для реализации методов глубокого обучения
— Реализован пример анализа текстов (на примере дата сета из открытых источников)

В результате работы было установлено, что глубокое обучение нейронных сетей имеет огромный потенциал для создания инновационных продуктов и решения сложных задач в различных областях, таких как медицина, финансы, транспорт и многие другие.

В процессе выполнения дипломной работы решалась задача классификации текстов новостей. Было проведено глубокое обучение трех нейронных сетей с последующей оценкой их эффективности.

Основные источники информации:

1. «Логические исчисления идей, относящиеся к нервной деятельности » Уоррен Маккалок, Уолтер Питтс
2. «Алгоритмы Data Science и их практическая реализация на Python» Протодьяконов, Пылов, Садовников
3. Статья «Python и машинное обучение»
4. «Глубокое обучение на Python» Питер Франсуа Шолле
5. Aurelien Geron.Прикладное машинное обучение с помощью Scikit-Learn, Keras и TensorFlow
6. Andrew Tusk. Grokking Deep Learning
7. Denis Shaikhislamov, Andrey Sozykin, Vadim Voevodin. Survey on software tools that implement deep learning algorithms on intel/x86 and IBM/Power8/Power platforms // Supercomputing Frontiers and Innovations.
8. Adrian Rosebrock. Deep Learning for Computer Vision with Python

9. Maxim Lapan. Deep Reinforcement Learning Hands-On
10. Сергей Николенко, А. Кадурин, Екатерина Архангельская. Глубокое обучение. Погружение в мир нейронных сетей