ФЕДЕРАЛЬНОЕ АГЕНСТВО СВЯЗИ

Ордена Трудового Красного Знамени федеральное государственное бюджетное образовательное учреждение высшего образования

Московский технический университет связи и информатики

(МТУСИ)

Кафедра «Математическая кибернетика и информационные технологии» Дисциплина «Интеллектуальные системы»

Лабораторная работа № 7

«Классификация обзоров фильмов»

Выполнил: \_\_\_\_\_\_\_ Д.С. Калининский

Проверил: \_\_\_\_\_\_\_ А.В. Шереметьев

Москва 2022

**Оглавление**

[1 Цель работы 3](#_Toc119580633)

[2 Задачи 3](#_Toc119580634)

[3 Выполнение работы 3](#_Toc119580635)

[4 Результаты 5](#_Toc119580636)

[5 Вывод 7](#_Toc119580637)

# **1 Цель работы**

Классификация последовательностей - это проблема прогнозирующего моделирования, когда у вас есть некоторая последовательность входных данных в пространстве или времени, и задача состоит в том, чтобы предсказать категорию для последовательности.

Проблема усложняется тем, что последовательности могут различаться по длине, состоять из очень большого словарного запаса входных символов и могут потребовать от модели изучения долгосрочного контекста или зависимостей между символами во входной последовательности.

В данной лабораторной работе также будет использоваться дата сет IMDb, однако обучение будет проводиться с помощью рекуррентной нейронной сети.

# **2 Задачи**

1) Ознакомиться с рекуррентными нейронными сетями;

2) Изучить способы классификации текста;

3) Ознакомиться с ансамблированием сетей;

4) Построить ансамбль сетей, который позволит получать точность не менее 97%.

# **3 Выполнение работы**

Начнем с импорта необходимых зависимостей для предварительной обработки данных и построения модели (рисунок 1).

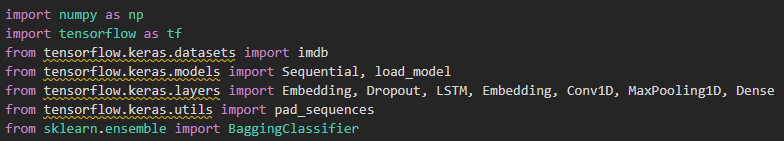


Рисунок 1 – Импорт необходимых модулей

Загрузим датесет IMDb, который уже встроен в Keras. Поскольку мы не хотим иметь данные обучения и тестирования в пропорции 50/50, мы сразу же объединим эти данные после загрузки для последующего разделения в пропорции 80/20 (рисунок 2)

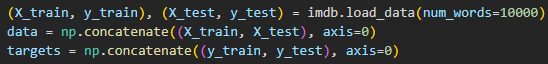


Рисунок 2 – Загрузка дата сета

Далее нам нужно обрезать и дополнить входные последовательности так, чтобы они были одинаковой длины для моделирования. Модель узнает, что нулевые значения не содержат никакой информации, поэтому на самом деле последовательности имеют разную длину с точки зрения содержания, но для выполнения вычислений в Керасе требуются векторы одинаковой длины (рисунок 3).

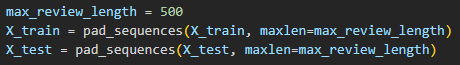


Рисунок 3 – Представляем в виде векторов одинаковой длины

Архитектура сети первый слой - это Embedded, который использует 32 вектора длины для представления каждого слова. Следующий уровень - это слой LSTM с 100 единицами памяти (умными нейронами). Наконец, поскольку это проблема классификации, мы используем плотный выходной слой с одним нейроном и сигмоидной функцией активации, чтобы сделать 0 или 1 прогноз для двух классов (хорошего и плохого) в задаче.

Поскольку это проблема двоичной классификации, в качестве функции потерь используется журнал потерь (binary\_crossentropy в Керасе). Используется эффективный алгоритм оптимизации ADAM. Модель подходит только для 2 эпох, потому что она быстро решает проблему. Большой пакет из 64 обзоров используется для разметки обновлений веса (рисунок 4).

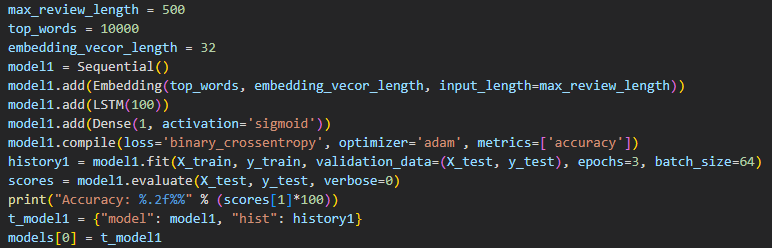


Рисунок 4 – Модель 1

Вы можете видеть, что этот простой LSTM с небольшими настройками достигает почти современных результатов по проблеме IMDB. Важно, что это шаблон, который вы можете использовать для применения сетей LSTM к вашим собственным задачам классификации последовательностей.

Рассмотрим некоторые расширения этой простой модели.

LSTM и сверточная нейронная сеть для классификации последовательностей. Сверточные нейронные сети превосходно изучают пространственную структуру во входных данных.

Данные обзора IMDB действительно имеют одномерную пространственную структуру в последовательности слов в обзорах, и CNN, может быть, в состоянии выбрать инвариантные особенности для хорошего и плохого настроения. Эти изученные пространственные особенности могут затем быть изучены как последовательности уровнем LSTM.

Мы можем легко добавить одномерный слой CNN и максимальный пул после слоя Embedding, которые затем передают объединенные элементы в LSTM. Мы можем использовать небольшой набор из 32 объектов с небольшой длиной фильтра 3. Слой пула может использовать стандартную длину 2, чтобы вдвое уменьшить размер карты объектов (рисунок 5).

Например, мы бы создали модель следующим образом:

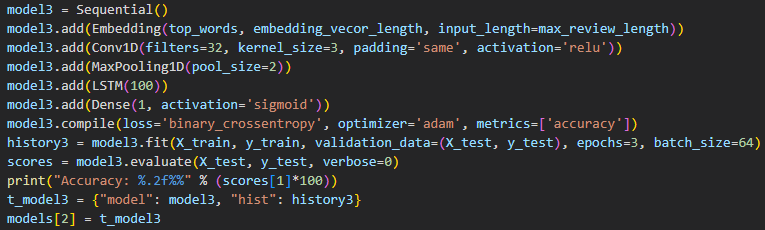


Рисунок 5 – Модель 3

Мы видим, что мы достигаем результатов, аналогичных первому примеру, но с меньшими весами и меньшим временем обучения.

Рекуррентные нейронные сети, такие как LSTM, обычно имеют проблему переобучения.

Вам необходимо решить эту проблему путем добавления в архитектуру сети слоев Dropout (рисунок 6-7).

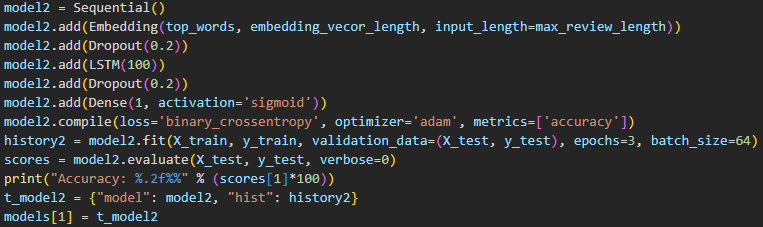


Рисунок 6 – Модель 2

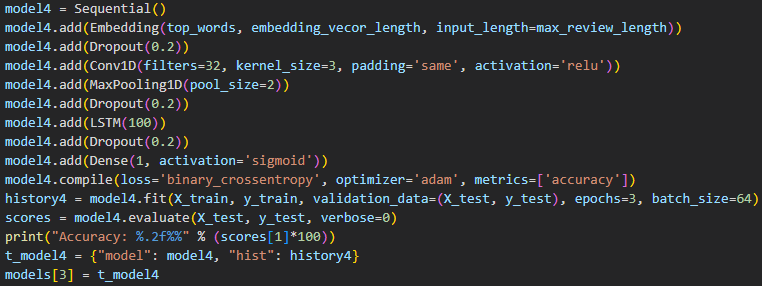


Рисунок 7 – Модель 4

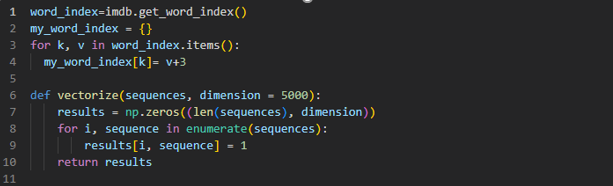
# **4 Результаты**

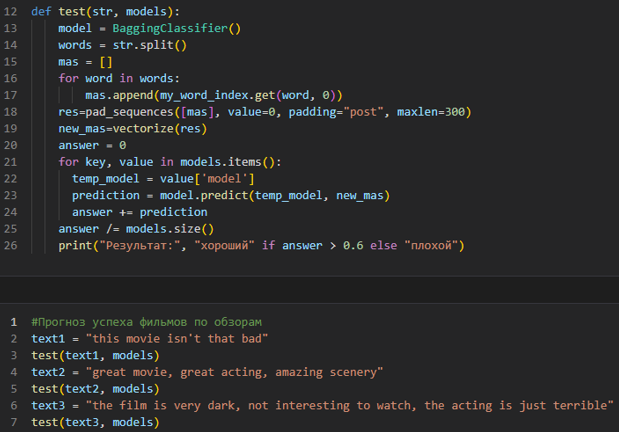
Данной лабораторной работа состоит из 4 моделей. Результаты тестирования моделей в таблице 1.

Таблица 1 – Результаты тестирования моделей

|  |  |
| --- | --- |
| Модель | Точность |
| 1 | 85.2% |
| 2 | 87.1% |
| 3 | 86.9% |
| 4 | 87.7% |

Исходя из полученных результатов делаем вывод, что четвёртая модель является самой оптимальной так, как имеет наивысшую точность. Далее нужно было провести ансамблирование моделей и написать функцию/функции, которые позволят загружать текст и получать результат ансамбля сетей (рисунок 8).



Рисунок 8 – Ансамблирование моделей

Результаты тестирование ансамблирование моделей приведены в таблице 2.

Таблица 2 – Результаты тестирования фраз

|  |  |
| --- | --- |
| Текст | Результат |
| this movie isn't that bad | Хороший |
| great movie, great acting, amazing scenery | Хороший |
| the film is very dark, not interesting to watch, the acting is just terrible | Плохой |

# **5 Вывод**

В данной лабораторной работе, была реализована Классификация обзоров фильмов. Для реализации задачи было построено 4 модель и протестированы для выявления лучшей. Так же было выполнено ансамблирование моделей и написана функция для проверки выражений пользователя.