МИНОБРНАУКИ РОССИИ
САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ
ЭЛЕКТРОТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
«ЛЭТИ» ИМ. В.И. УЛЬЯНОВА (ЛЕНИНА)
Кафедра МО ЭВМ

ОТЧЕТ

по лабораторной работе №6
по дисциплине «Искусственные нейронные сети»
Тема: Прогноз успеха фильмов по обзорам

Студент гр. 8382 _____ Гордиенко А.М.

Преподаватель _____ Жангиров Т.Р.

Санкт-Петербург
2021

Цель работы.

Прогноз успеха фильмов по обзорам (Predict Sentiment From Movie Reviews).

Порядок выполнения работы.

- Ознакомиться с задачей классификации
- Изучить способы представления текста для передачи в ИНС
- Достигнуть точность прогноза не менее 95%

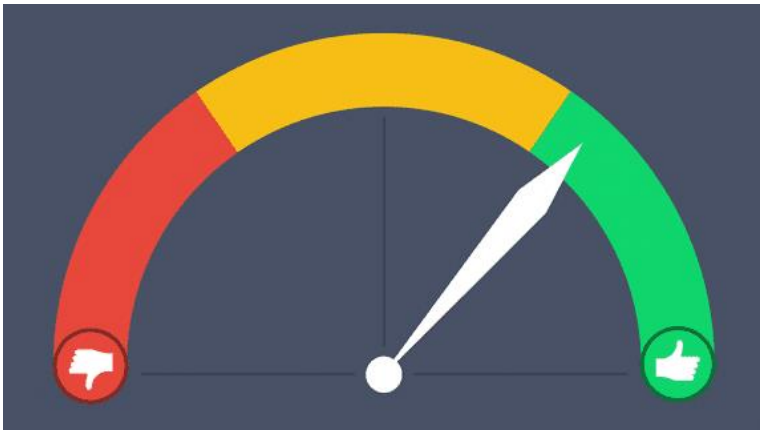
Требования.

1. Построить и обучить нейронную сеть для обработки текста
2. Исследовать результаты при различном размере вектора представления текста
3. Написать функцию, которая позволяет ввести пользовательский текст (в отчете привести пример работы сети на пользовательском тексте)

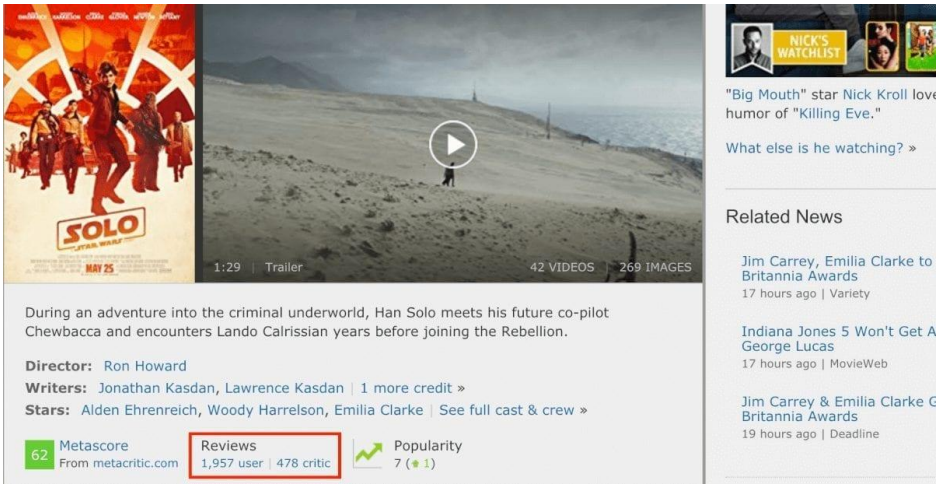
Основные теоретические положения.

Что такое анализ настроений (сентимент-анализ)?

С помощью анализа настроений можно определить отношение (например, настроение) человека к тексту, взаимодействию или событию. Поэтому сентимент-анализ относится к области обработки естественного языка, в которой смысл текста должен быть расшифрован для извлечения из него тональности и настроений.



Спектр настроений обычно подразделяется на положительные, отрицательные и нейтральные категории. С использованием анализа настроений можно, например, прогнозировать мнение клиентов и их отношение к продукту на основе написанных ими обзоров. Поэтому анализ настроений широко применяется к обзорам, опросам, текстам и многому другому.



Датасет IMDb состоит из 50 000 обзоров фильмов от пользователей, помеченных как положительные (1) и отрицательные (0).

- Рецензии предварительно обрабатываются, и каждая из них кодируется последовательностью индексов слов в виде целых чисел.
- Слова в обзорах индексируются по их общей частоте появления в датасете. Например, целое число «2» кодирует второе наиболее частое используемое слово.
- 50 000 обзоров разделены на два набора: 25 000 для обучения и 25 000 для тестирования.

Датасет был создан исследователями Стэнфордского университета и представлен в статье 2011 года, в котором достигнутая точность предсказаний была равна 88,89%. Датасет также использовался в рамках конкурса сообщества Kaggle «Bag of Words Meets Bags of Popcorn» в 2011 году.

Ход работы.

Подключение необходимых зависимостей:

```
import matplotlib
```

```
import matplotlib.pyplot as plt
import numpy as np
from keras.utils import to_categorical
from keras import models
from keras import layers
```

Загрузка и обработка датасета:

```
from keras.datasets import imdb
(training_data, training_targets), (testing_data, testing_targets) = imdb.load_data(num_words=10000)
data = np.concatenate((training_data, testing_data), axis=0)
targets = np.concatenate((training_targets, testing_targets), axis=0)
```

Функция векторизации данных:

```
def vectorize(sequences, dimension = 10000):
    results = np.zeros((len(sequences), dimension))
    for i, sequence in enumerate(sequences):
    results[i, sequence] = 1
    return results

data = vectorize(data)
targets = np.array(targets).astype("float32")
```

Отзывы представлены в числовом закодированном формате. Для расшифровки используется следующая конструкция:

```
index = imdb.get_word_index()
reverse_index = dict([(value, key) for (key, value) in index.items()])
decoded = " ".join( [reverse_index.get(i - 3, "#") for i in data[0]] )
print(decoded)
```

Пример кодированного и декодированного отзыва:

[1, 14, 22, 16, 43, 530, 973, 1622, 1385, 65, 458, 4468, 66, 3941, 4, 173, 36, 256, 5, 25, 100, 43, 838, 112, 50, 670, 2, 9, 35, 480, 284, 5, 150, 4, 172, 112, 167, 2, 336,	# this film was just brilliant casting location scenery story direction everyone's really suited the part they played and you could just imagine being

385, 39, 4, 172, 4536, 1111, 17, 546, 38, 13, 447, 4, 192, 50, 16, 6, 147, 2025, 19, 14, 22, 4, 1920, 4613, 469, 4, 22, 71, 87, 12, 16, 43, 530, 38, 76, 15, 13, 1247, 4, 22, 17, 515, 17, 12, 16, 626, 18, 2, 5, 62, 386, 12, 8, 316, 8, 106, 5, 4, 2223, 5244, 16, 480, 66, 3785, 33, 4, 130, 12, 16, 38, 619, 5, 25, 124, 51, 36, 135, 48, 25, 1415, 33, 6, 22, 12, 215, 28, 77, 52, 5, 14, 407, 16, 82, 2, 8, 4, 107, 117, 5952, 15, 256, 4, 2, 7, 3766, 5, 723, 36, 71, 43, 530, 476, 26, 400, 317, 46, 7, 4, 2, 1029, 13, 104, 88, 4, 381, 15, 297, 98, 32, 2071, 56, 26, 141, 6, 194, 7486, 18, 4, 226, 22, 21, 134, 476, 26, 480, 5, 144, 30, 5535, 18, 51, 36, 28, 224, 92, 25, 104, 4, 226, 65, 16, 38, 1334, 88, 12, 16, 283, 5, 16, 4472, 113, 103, 32, 15, 16, 5345, 19, 178, 32]	there robert # is an amazing actor and now the same being director # father came from the same scottish island as myself so i loved the fact there was a real connection with this film the witty remarks throughout the film were great it was just brilliant so much that i bought the film as soon as it was released for # and would recommend it to everyone to watch and the fly fishing was amazing really cried at the end it was so sad and you know what they say if you cry at a film it must have been good and this definitely was also # to the two little boy's that played the # of norman and paul they were just brilliant children are often left out of the # list i think because the stars that play them all grown up are such a big profile for the whole film but these children are amazing and should be praised for what they have done don't you think the whole story was so lovely because it was true and was someone's life after all that was shared with us all

1. Построить и обучить нейронную сеть для обработки текста.

Модель сети имеет следующую архитектуру:

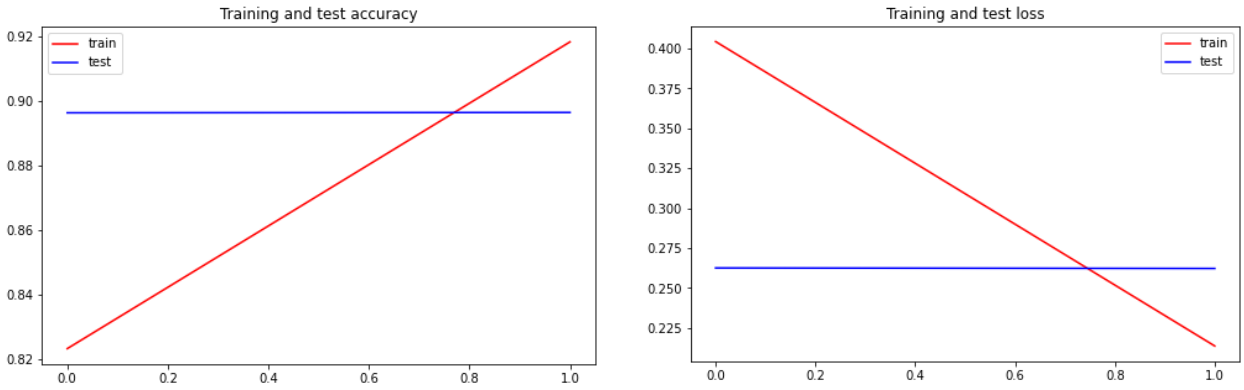
```
model = models.Sequential()
model.add(layers.Dense(50, activation="relu", input_shape=(dimension,)))
model.add(layers.Dropout(0.4, noise_shape=None, seed=None))
model.add(layers.Dense(30, activation="relu"))
model.add(layers.Dropout(0.4, noise_shape=None, seed=None))
```

```
model.add(layers.Dense(30, activation="relu"))
model.add(layers.Dense(1, activation="sigmoid"))
```

Построенная модель была собрана и обучена при следующих параметрах.

```
model.compile(optimizer="adam", loss="binary_crossentropy",
              metrics=["accuracy"])
h = model.fit(train_x, train_y, epochs=2, batch_size=500,
              validation_data=(test_x, test_y))
```

Графики точности и ошибки представлены на рис. 1.



а б

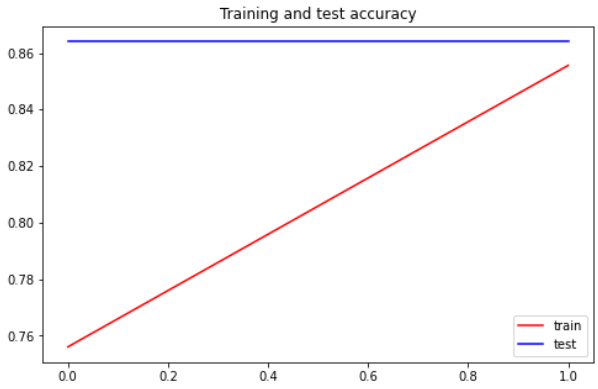
Рисунок 1 – График точности(а) и ошибки(б).

2. Исследовать результаты при различном размере вектора представления текста.

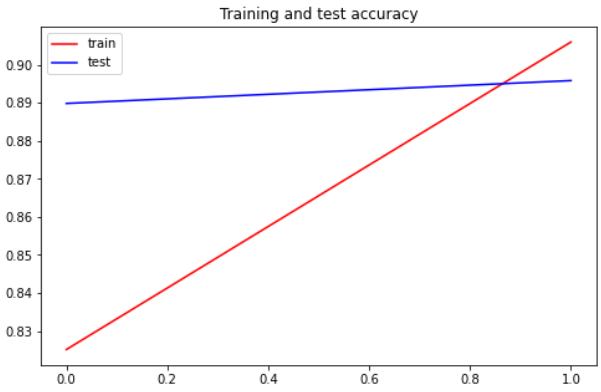
Графики точности представлены на рис. 2. Значения представлены в таблице 1.

dimension	Значение точности
1000	0.8640
5000	0.8928
10000	0.89585
15000	0.89283

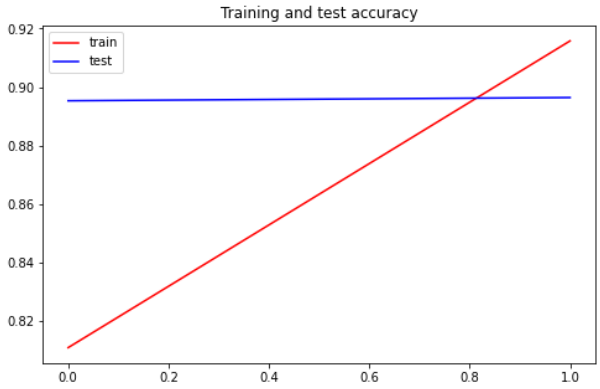
Таблица 1.



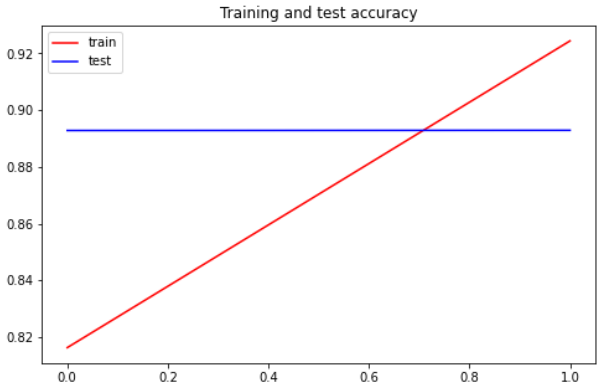
dimensions = 1000



dimensions = 5000



dimensions =10000



dimensions = 15000

Рисунок 2 – Графики точности при различных размерах словаря.

По результатам видно, что при увеличении словаря увеличивается точность модели, но увеличивается вероятность переобучения а также нагрузка на ОЗУ, что не дало возможность оценить работу сети при размере словаря в 20000.

3. Написать функцию, которая позволяет ввести пользовательский текст.

Была написана функция load_user_input, описаная следующим образом:

```
def load_user_input(text, target):
   def genNum(data, dic):
       data                                                                    =
data.translate(str.maketrans(dict.fromkeys(string.punctuation))).split()
       for i in range(len(data)):
        num = dic.get(data[i])
       if (num == None):
                data[i] = 0
            else:
```

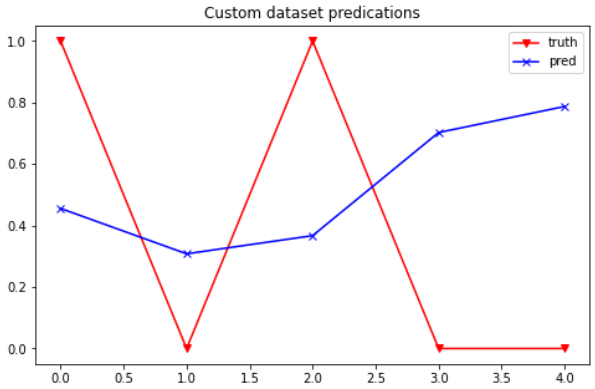
```
                data[i] = num
        return data

  dic = dict(datasets.imdb.get_word_index())
  test_x = []
 test_y = np.array(target).astype("float32")
 for i in range(0, len(text)):
    test_x.append(genNum(text[i], dic))
 test_x = vectorize(test_x)
return test_x, test_y
```

Пример пользовательских данных:

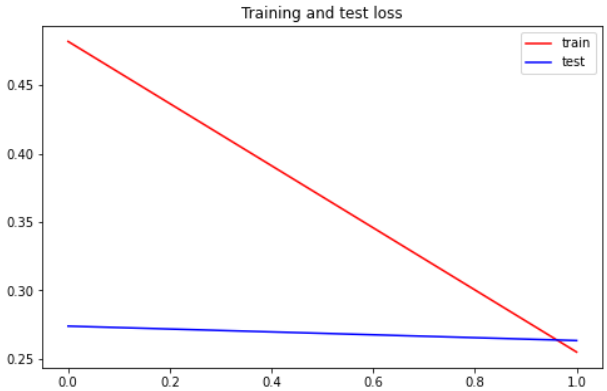
```
x = [
        "Amazing film love this genre",
        "Storytelling felt primitive and kinda forced",
        "Absolutely astonishing bravo",
        "Could not cope with pacing of the plot just bad",
        "This artwork did a bare minimum that is not enough"
]
y = [1, 0, 1, 0, 0]
```

Точность составила 0.89.

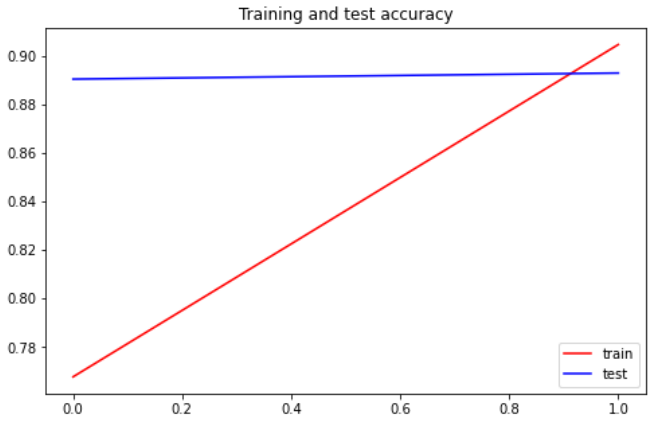
Результат работы сети с пользовательским вводом представлен на рис. 3.



Предсказание



Потери



Точность

Рисунок 3 – Графики предсказания, потери и точности.

Выводы.

В ходе работы была изучена задача классификации настроений на примере отзывов о фильмах. Было изучено как передавать текстовые данные искусственной нейронной сети, как влияет размер словаря на результаты работы сети. Была получена сеть, имеющая точность около 90%. Из эксперимента с пятью неизвестными для сети отзывами, следует, что сеть корректно определяет настроение текста.