**МИНОБРНАУКИ РОССИИ**

**Санкт-Петербургский государственный**

**электротехнический университет**

**«ЛЭТИ» им. В.И. Ульянова (Ленина)**

**Кафедра математического обеспечения и применения ЭВМ**

отчет

**по лабораторной работе №7**

**"Классификация обзоров фильмов"**

**по дисциплине «Искусственные нейронные сети»**

|  |  |  |
| --- | --- | --- |
| Студент гр. 8383 |  | Бабенко Н.С. |
| Преподаватель |  | Жангиров Т.Р. |

Санкт-Петербург

2021

## Цель.

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 датасет IMDb, однако обучение будет проводиться с помощью рекуррентной нейронной сети.

## Задание.

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%
5. Найти набор оптимальных ИНС для классификации текста
6. Провести ансамблирование моделей
7. Написать функцию/функции, которые позволят загружать текст и получать результат ансамбля сетей
8. Провести тестирование сетей на своих текстах (привести в отчете)

## Выполнение работы.

В программе было реализовано две модели. Листинг первой модели:

model = Sequential()

model.add(Embedding(10000, 32, input\_length=500))

model.add(Conv1D(filters=32, kernel\_size=3, padding='same', activation='relu'))

model.add(Dropout(0.3))

model.add(MaxPooling1D(pool\_size=2))

model.add(Dropout(0.5))

model.add(LSTM(100))

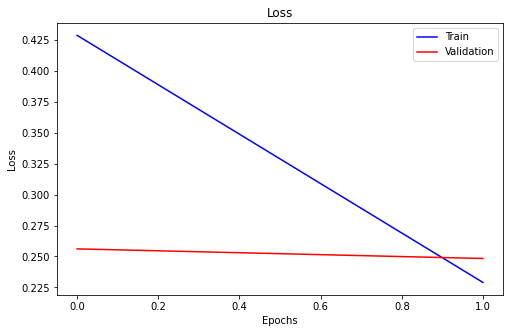
model.add(Dropout(0.3))

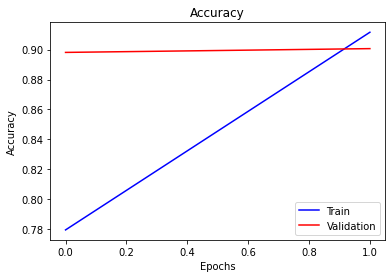
model.add(Dense(1, activation='sigmoid'))

model.compile(loss='binary\_crossentropy', optimizer='adam', metrics=['accuracy'])

model.fit(x\_train, y\_train, batch\_size=128, epochs=2, verbose=1, validation\_split=0.1)

Результаты тестирования первой модели:





Точность на тестовых данных составила 0.9016

Листинг второй модели:

model\_2 = Sequential()

model\_2.add(Embedding(10000, 32, input\_length=500))

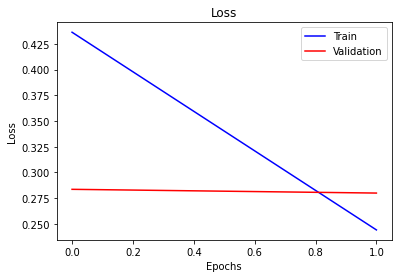
model\_2.add(LSTM(100))

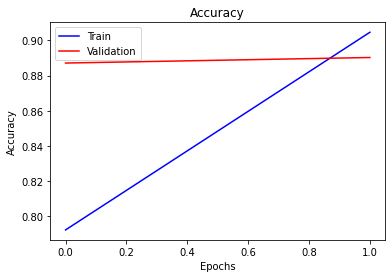
model\_2.add(Dense(1, activation='sigmoid'))

model\_2.fit(x\_train, y\_train, batch\_size=128, epochs=2, verbose=1, validation\_split=0.1)

model\_2.compile(loss='binary\_crossentropy', optimizer='adam', metrics=['accuracy'])

Результаты тестирования второй модели:





Точность на тестовых данных составила 0.8912

Первая модель, которая помимо LSTM имеет сверточные слои, показала лучшую точность, чем вторая без свертки.

На пользовательских тестах был проверен ансамбль моделей (ансамблирование путем выбора среднего из двух предсказанных значений моделей)

Текст: "It is bad, I hate it"

Предсказание ансамбля: 0.324

Текст: "Really boring and bad, I am dissapointed"

Предсказание ансамбля: 0.4765

Текст: "It's amazing, fantastic and exiting"

Предсказание ансамбля: 0.62278

Текст: "Really good, fantastic actors, I like it"

Предсказание ансамбля: 0.6466

Ансамбль моделей корректно предсказал настроения обзоров на фильмы.

## Выводы.

В ходе выполнения лабораторной работы было изучено решение задачи классификации отзывов на фильмы для определения успешности фильма с помощью рекуррентных сетей. Был создан и протестирован ансамбль из двух моделей.