

Санкт-Петербургский национальный исследовательский университет  
информационных технологий, механики и оптики

**Лабораторная работа №7**  
**Дисциплина «Разработка интеллектуальных систем»**

**Выполнил:**  
Съестов Дмитрий Вячеславович  
Группа Р3417

**Преподаватель:**  
Жукова Наталия Александровна

Санкт-Петербург  
2020

## Листинг программы

```
import numpy
from keras.datasets import imdb
from keras.models import Sequential
from keras.layers import Dense, LSTM, Conv1D, MaxPooling1D, Dropout
from keras.layers.embeddings import Embedding
from keras.preprocessing import sequence
import matplotlib.pyplot as plt

top_words = 5000
(X_train, y_train), (X_test, y_test) = imdb.load_data(num_words=top_words)

max_review_length = 500
X_train = sequence.pad_sequences(X_train, maxlen=max_review_length)
X_test = sequence.pad_sequences(X_test, maxlen=max_review_length)

embedding_vector_length = 32
model = Sequential()
model.add(Embedding(top_words, embedding_vector_length,
input_length=max_review_length))
model.add(Dropout(0.3, noise_shape=None, seed=None))
model.add(Conv1D(filters=32, kernel_size=3, padding='same',
activation='relu'))
model.add(MaxPooling1D(pool_size=2))
model.add(Dropout(0.2, noise_shape=None, seed=None))
model.add(LSTM(100))
model.add(Dense(1, activation='sigmoid'))

model.compile(
    optimizer = "adam",
    loss = "binary_crossentropy",
    metrics = ["accuracy"]
)

H = model.fit(
    X_train, y_train,
    epochs = 2,
    batch_size = 100,
    validation_data = (X_test, y_test)
)

loss = H.history['loss']
val_loss = H.history['val_loss']
acc = H.history['accuracy']
val_acc = H.history['val_accuracy']

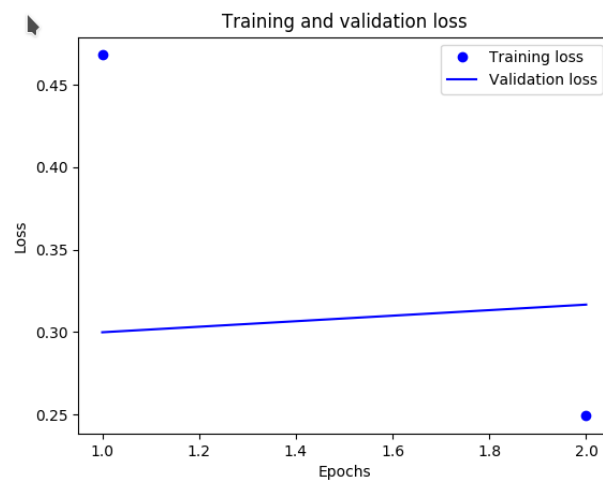
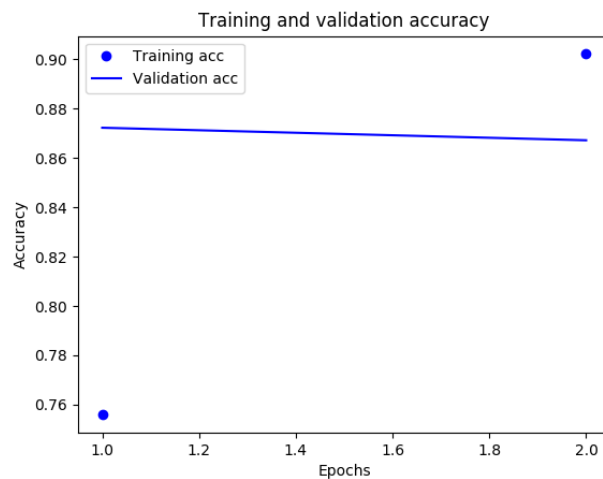
epochs = range(1, len(loss) + 1)

# Построение графика ошибки
plt.plot(epochs, loss, 'bo', label='Training loss')
plt.plot(epochs, val_loss, 'b', label='Validation loss')
plt.title('Training and validation loss')
plt.xlabel('Epochs')
plt.ylabel('Loss')
plt.legend()
plt.show()
```

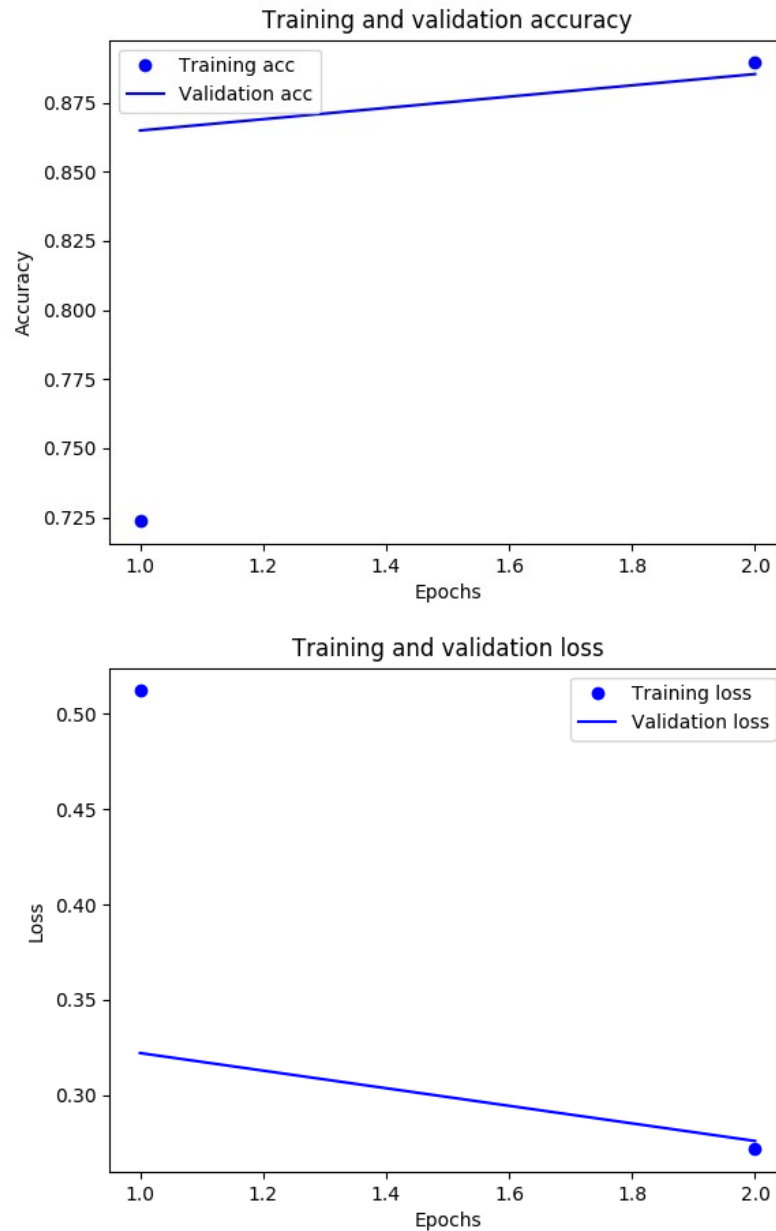
```
# Построение графика точности
plt.clf()
plt.plot(epochs, acc, 'bo', label='Training acc')
plt.plot(epochs, val_acc, 'b', label='Validation acc')
plt.title('Training and validation accuracy')
plt.xlabel('Epochs')
plt.ylabel('Accuracy')
plt.legend()
plt.show()
```

## Результаты

Без слоёв Dropout:



Со слоями Dropout:



### Вывод

Как видно по графикам, без новых слоёв происходило переобучение, о чём свидетельствует снижение точности и повышение потерь по мере обучения. Добавление двух дополнительных слоёв Dropout решило эту проблему.

