Министерство науки и высшего образования Российской Федерации
Калужский филиал федерального государственного автономного образовательного учреждения высшего образования
«Московский государственный технический университет имени Н.Э. Баумана
(национальный исследовательский университет)»
(КФ МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ ИУК «Информатика и управление»

КАФЕДРА ИУК5 «Системы обработки информации»

ДОМАШНЯЯ РАБОТА №2

«ПОСИМВОЛЬНАЯ ГЕНЕРАЦИЯ ТЕКСТА НА ОСНОВЕ LSTM»

по дисциплине: «Методы глубокого обучения»

Выполнил: студент группы ИУК5-21М _____ А. Э. Дармограй
(Подпись) (И.О. Фамилия)

Проверил: _____ Ю. С. Белов
(Подпись) (И.О. Фамилия)

Дата сдачи (защиты):

Результаты сдачи (защиты):

- Балльная оценка:
- Оценка:

Калуга, 2025

Цель работы:

Целью выполнения домашней работы является получение практических навыков в построении слоя LSTM при посимвольной генерации текста

Задачи:

1. Изучить алгоритм долгой краткосрочной памяти LSTM
2. Разработать модель глубокой нейронной сети со слоем LSTM
3. Применить алгоритм LSTM к посимвольной генерации текста

Выполнение работы

Код доступен в репозитории GitHub:

https://github.com/Dariarty/Deep_Learning_Methods

Данную работу выполнял на Python версии 3.9.13 и Tensorflow версии 2.7.0

Код домашней работы №2:

https://github.com/Dariarty/Deep_Learning_Methods/blob/main/src/DR_2/lstm.ipynb

Посимвольная генерация текста на основе LSTM

```
#В данной работе использую Python 3.9.13 и tensorflow 2.7.0

import sys
import tensorflow as tf
from tensorflow import keras

# Вывод версий Python и Tensorflow
print("Python", sys.version)
print("Tensorflow", tf.__version__)

# Убеждаюсь, что tensorflow использует GPU
available_gpus = tf.config.list_physical_devices('GPU') # Динамическое использование памяти GPU
if available_gpus:
    try:
        for gpu in available_gpus:
            tf.config.experimental.set_memory_growth(gpu, True)
        print("Tensorflow uses GPU")
    except RuntimeError as error:
        print("GPU Error:", error)
```

```
Python 3.9.13 (tags/v3.9.13:6de2ca5, May 17 2022, 16:36:42) [MSC v.1929 64 bit (AMD64)]
Tensorflow 2.7.0
Tensorflow uses GPU
```

Загрзука и подготовка данных

Будем обучать модель на романе "Гордость и предубеждение" на английском языке. Переходы на новую строку заменены пробелами, все

буквы приведены в младший регистр, убраны служебные отметки в квадратных скобках.

```
import numpy as np
import re

path = keras.utils.get_file('pride_and_prejudice.txt',
origin='https://www.gutenberg.org/files/1342/1342-0.txt')

#Обработка текста
text = open(path, encoding='utf-8').read().lower()
text = text.replace('\n', ' ') # Заменяю переходы на новую строку
пробелами

#Отбрасываем оглавление и служебную информацию
start = text.find("preface.")
end = text.find("end of the project gutenberg ebook")
text = text[start:end]

#Отбрасываем служебные отметки в кавычках [...]
text = re.sub(r'\[.*?\]', '', text)

#Убираем множественные пробелы
text = re.sub(r'\s+', ' ', text)

print('Corpus length:', len(text))
```

Corpus length: 712148

Векторизация последовательностей символов

Разбиваем текст на частично перекрывающиеся последовательности фиксированной длины с шагом 3 символа. Формируются входные последовательности и целевые символы

```
# Извлечение последовательностей по 60 символов
maxlen = 60

# Новые последовательности выбираются через каждые 3 символа
step = 3

# Хранение извлеченных последовательностей
sentences = []

#Хранение целей (символов, следующих за последовательностями)
next_chars = []
for i in range(0, len(text) - maxlen, step):
    sentences.append(text[i: i + maxlen])
    next_chars.append(text[i + maxlen])
print('Number of sequences:', len(sentences))
```

```
# Список уникальных символов в корпусе
chars = sorted(list(set(text)))
print('Unique characters:', len(chars))

#Словарь,отображающий уникальные символы в их индексы в списке «chars»
char_indices = dict((char, chars.index(char)) for char in chars)
print('Vectorization...')

x = np.zeros((len(sentences), maxlen, len(chars)), dtype=bool)
y = np.zeros((len(sentences), len(chars)), dtype=bool)

for i, sentence in enumerate(sentences):
    for t, char in enumerate(sentence):
        x[i, t, char_indices[char]] = 1
    # Прямое кодирование символов в бинарные массивы
    y[i, char_indices[next_chars[i]]] = 1
```

```
Number of sequences: 237363
Unique characters: 59
Vectorization...
```

Создание модели

Создаётся модель из слоя LSTM (128 нейронов) и выходного полносвязного слоя с функцией активации softmax. Используется оптимизатор RMSprop с повышенным learning_rate=0.01

```
model = keras.models.Sequential()
model.add(keras.layers.LSTM(128,input_shape=(maxlen,len(chars))))
model.add(keras.layers.Dense(len(chars), activation='softmax'))

# Конфигурация компилируемой модели
optimizer = keras.optimizers.RMSprop(learning_rate=0.01)
model.compile(loss='categorical_crossentropy', optimizer=optimizer)
```

Функция выбора символа с температурой

Преобразует вероятности, предсказанные моделью, с учётом параметра температуры. Чем выше температура, тем выше случайность в выборе следующего символа.

```
def sample(preds, temperature=1.0):
    preds = np.asarray(preds).astype('float64')
    preds = np.log(preds + 1e-8) / temperature #Добавляю 1e-8 чтобы
избежать логарифма от нуля в случае preds==0
    exp_preds = np.exp(preds)
    preds = exp_preds / np.sum(exp_preds)
    probas = np.random.multinomial(1, preds, 1)
    return np.argmax(probas)
```

Обучение модели и генерация текста

В течение 60 эпох модель обучается на всём корпусе. После каждой эпохи случайным образом выбирается начальный фрагмент текста, и на его основе с помощью модели генерируется 400 символов текста для каждой из температур: 0.2, 0.5, 1.0 и 1.2. Результат генерации сохраняется в файл по эпохам.

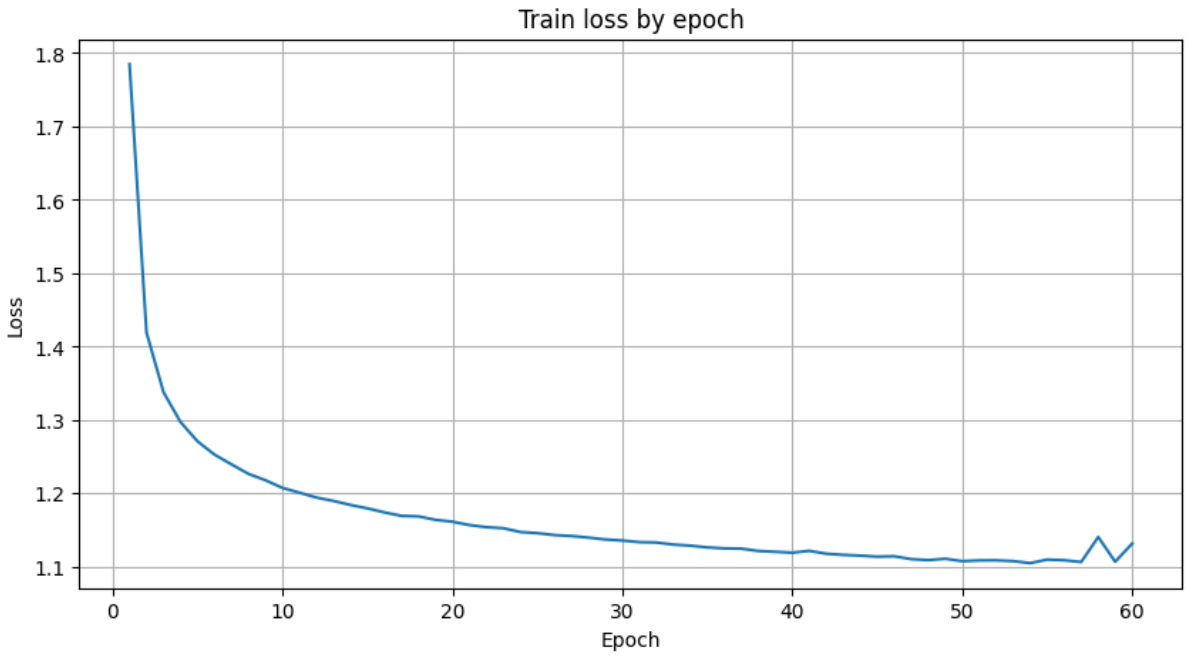
```
import random
import sys
import os

#Директория для сохранения сгенерированных текстов
OUTPUT_DIR = "generated_text"
os.makedirs(OUTPUT_DIR, exist_ok=True)

#Количество эпох
EPOCHS = 60

#Будем сохранять потери
losses = []

# Обучение модели
for epoch in range(1, EPOCHS + 1):
    print('Epoch', epoch)
    #Выполнение одной итерации обучения
    history = model.fit(x, y, batch_size=128, epochs=1)
    losses.append(history.history['loss'][0])

    # Выбор случайного начального текста
    start_index = random.randint(0, len(text) - maxlen - 1)
    seed_text = text[start_index: start_index + maxlen]

    #Открытие файла для записи
    filename = f"epoch_{epoch:02}.txt"
    filepath = os.path.join(OUTPUT_DIR, filename)

    with open(filepath, "w", encoding="utf-8") as file:
        print("Generating to file:", filename)
        print()
        file.write(f"Epoch: {epoch}\n\n")
        file.write(f"Generating with seed:\n{seed_text}\n\n")

        # Генерация текста для разных температур
        for temperature in [0.2, 0.5, 1.0, 1.2]:
            full_gen = seed_text #Полный сгенерированный текст для записи
в файл
            generated_text = seed_text

            # Генерация 400 символов, начиная с начального текста
            for i in range(400):
            # Прямое кодирование символов, сгенерированных до сих пор
                sampled = np.zeros((1, maxlen, len(chars)))
                for t, char in enumerate(generated_text):
```

```
                    sampled[0, t, char_indices[char]] = 1.

                # Выбор следующего символа
                preds = model.predict(sampled, verbose=0)[0]
                next_index = sample(preds, temperature)
                next_char = chars[next_index]

                generated_text += next_char
                generated_text = generated_text[1:]
                full_gen += next_char

            file.write("-"*50 + "\n")
            file.write(f"Temperature: {temperature}\n\n")
            file.write(full_gen + "\n\n")

model.save("char_lstm_model.h5")
```

Визуализация изменения функции потерь

train_loss по эпохам обучения

```
import matplotlib.pyplot as plt

plt.figure(figsize=(10, 5))
plt.plot(range(1, len(losses) + 1), losses)
plt.title("Train loss by epoch")
plt.xlabel("Epoch")
plt.ylabel("Loss")
plt.grid(True)
plt.show()
```



Видим, что потери постепенно снижаются примерно до 1.1. На 58-й эпохе наблюдается локальный рост значения функции потерь. Это может быть

связано с влиянием отдельного батча, числовой нестабильностью или недостаточно низким значением параметра learning rate. При этом общее поведение модели остаётся корректным, и генерация текста не деградирует.

Пример генерации:

Epoch: 56

Generating with seed:

_ begun.” “my beauty you had early withstood, and as for my

Temperature: 0.2

_ begun.” “my beauty you had early withstood, and as for my dear sir willia some real expression of his considerations were soon as it was a sisters were spoken at all.” mr. bennet and such a great deal design of his sister was so silent of the character, and they wished him to her as it was to be sister to be already said a great deal of the character of his sisters were seeing him there was since you are not to be so one of the last of the character, who

---

Temperature: 0.5

_ begun.” “my beauty you had early withstood, and as for my dear sir you have nother it will be so expected to have the sisters must be designed to concern that they discover, she thought of her father see, a few days been there has nothing in the way. but it was such an account in the confession, they wished her that she received the time of their being so of the great deal at all.” mr. bennet was love to concern to his having have been so distressed by t

---

Temperature: 1.0

_ begun.” “my beauty you had early withstood, and as for my dlarcy,” replied perhapse them. “i just of it, but what she could think a very match; and elizabeth, readily individer laterly with this, and her shaon, your lady catherinan.” “what except a hours toteness that she were son it was, good both with comfrided from the mistake of her discovery-more theor conhidered to dear is not very turn, because wave a dillunion of their general comfort, though aad

---

Temperature: 1.2

_ begun.” “my beauty you had early withstood, and as for my dear craduit me, who everyliwing interpord, she thought you should tnot reaute, bingley tratesed, you marry been followed of lookxer, and it were seltested, in eye of such congratefunies contempt from home, again slarged to be tooh. visit, though very dify that mrs. bennet, when yet, and she soon certain such affairs without stail of withoutêjones. to like my sintin what ble wickham, voiced them f

Epoch: 59

Generating with seed:

youngest of all is lately married, and my eldest is somewhe

Temperature: 0.2

youngest of all is lately married, and my eldest is somewhers to be the wish of the subject, they are not a word of her mother’s subject was not to be all the proposals of the first receiving the wish of the subject of the subject, and with the serious attention of the world, they were to be all the strong observed of the family and such a seriously to be all the first recollection of the subject with the little as to be the subject of the first surprise

---

Temperature: 0.5

youngest of all is lately married, and my eldest is somewheable of happiness to be slent the habding the senting from the feelings of the morrowe of mr. bingley thought soon as he was done to be all my sister’s persons of the gentleman was delightful to the face for her affection of the attentions were all her evening which my own doing, we thank, whatever they who said, and what had been that her father, and as to be a so out of the character; and that w

---

Temperature: 1.0

youngest of all is lately married, and my eldest is somewher, me is my efttal happiness town, that take the evening, for an abshence, being very likely, however, lizzy, i tell the to the windows i have been believed, met her daughters,

for as this; and he was prefered him to speak well me, when she became you to elizabeth looked at her mind archoating by the denty, with their justicg at the idek with most to blame it property distance?" "yes, mr. darcy, a

---

Temperature: 1.2

youngest of all is lately married, and my eldest is somewhet bende," replied elizabeth, keling, as im felt they was integaod:gation; and he on hall olll doanne of an attention! was earle elizse"py?" "he isneinfbed all thanking bingley, much short," said no girt, he had freat her man-instish bingley; farce off feel of your daughter, and soliary, without one what _he_ have sees abodemedon early turnack of mr. bingley, wene found whatly tlem?" , well jane sh

Как видим, наиболее интересный текст получается при значении температуры 0.5

Вывод: в ходе выполнения лабораторной работы были сформированы практические навыки по построению слоя LSTM при посимвольной генерации текста. Были изучен алгоритм долгой краткосрочной памяти LSTM, а также разработана модель глубокой нейронной сети со слоем LSTM.