

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

ЛАБОРАТОРНАЯ РАБОТА №8
по дисциплине «Искусственные нейронные сети»
Тема: «Генерация текста на основе «Алисы в стране чудес»»

Студент гр. 7381

Дорох С.В.

Преподаватель

Жукова Н. А.

Санкт-Петербург

2020

Цель работы.

Рекуррентные нейронные сети также могут быть использованы в качестве генеративных моделей.

Это означает, что в дополнение к тому, что они используются для прогнозных моделей (создания прогнозов), они могут изучать последовательности проблемы, а затем генерировать совершенно новые вероятные последовательности для проблемной области.

Подобные генеративные модели полезны не только для изучения того, насколько хорошо модель выявила проблему, но и для того, чтобы узнать больше о самой проблемной области.

Задачи.

- Ознакомиться с генерацией текста
- Ознакомиться с системой Callback в Keras

Требования.

1. Реализовать модель ИНС, которая будет генерировать текст
2. Написать собственный CallBack, который будет показывать то, как генерируется текст во время обучения (то есть раз в какое-то количество эпох генерировать и выводить текст у необученной модели)
3. Отследить процесс обучения при помощи TensorFlowCallBack, в отчете привести результаты и их анализ

Ход работы.

1. Была реализована модель ИНС для генерации текста. Архитектура данной модели совпадает с архитектурой модели в методических указаниях.

2. Был создан класс CustomCallback. Данный класс наследуется от класса `keras.callbacks.Callback`. В данном классе был переопределён

метод `on_epoch_end`, который в конце каждой эпохи, вызывает функцию, которая генерирует входные данные и подает их на вход недообученной модели. Код класса и функции имеется в приложении.

3. Ниже будут представлены тексты, сгенерированные моделью на различных эпохах.

Seed:

" fit--" you never had fits, my dear, i think?' he said to the queen.'never!' said the queen furiou "

aa a

Выше представлен текст после первой эпохи. Было сгенерировано всего два слова с одной и той же буквой.

Seed: " ed her back to the croquet-ground. the other guests had taken advantage of the queen's absence, and "

the soied to the toiee and the woied to tee thet she was aol aalin to the toeee the har ho a lott oi the toeee and the woied to the toiee and the woied to tee thet she was aol aalit toe toeee to the thet so tee the harter and the woied to the toiee tf the toeee tfe thit har sene the was soiee an the woiee 'no toe tae ' she mant hn a gort of the more, ''the more to tee to tee the more tf the toeee '

'i whan toe karter a soie,' said alice. ''the woued toe toetet to tee the more tf the toeee '

'i whan the woied to tee ' said the moee turtle. ''thel i toen the woied to tee to tee thet ' she said to herself, ''the more th the toiet ' she said to herself, ''the more th the toiet ' said the moee turtle. ''thel i toen the woied to tee to tee thet ' she said to herself, ''the more th the toiet ' she said to herself, ''the more th the toiet ' said the moee turtle. ''thel i toen the woied to tee to tee thet ' she said to herself, ''the more th the toiet ' she said to herself, ''the more

Выше представлен текст, сгенерированный на 10-й эпохе. Как можно заметить. В отличии от 1-й эпохи текст теперь представлен более, чем одним словом, но эти предложения в основной массе представляют пока набор непонятных и несвязных слов.

Seed:

" ething worth hearing. for some
minutes it puffed away without speaking, but at last it
unfolded its "
an in so the that she was so ae anl of the sooe, and the poeen
sas an the could, and she white rabbit wo ter the housh sas oo
the tabbit and the dould so ae anl of the sooe, and the poeen
sas an the could, and she white rabbit so ter the hough oo the
wan oo the tabbit and the whnted an the could so aedin she
would be a gotg oi the tooel of the soeee of the court,
and the whst on tere the gorse so tee that she was so ae an
fore an she could,
'the moeen su tete the dotmouse, said the daterpillar.

'whu, i shanl ie soen io the soeel to toe kooe, shi gatter
weit on a goea toiee an the coold,
'the dorstes shat io whe soeet ' she raid to herself, ''the
soeel th the sueee ' she mand to herself, ''the soeel th the
sueee ' she mand to herself, ''the soeel th the sueee ' she
mand to herself, ''the soeel th the sueee ' she mand to
herself, ''the soeel th the sueee ' she mand to herself, ''the
soeel th the sueee ' she mand to herself, ''the soeel th the
sueee ' she mand to herself, ''the s

Выше представлен текст, сгенерированный после 15-й эпохи. Как можно заметить стало меньше повторяющихся слов, а также стали чаще встречаться слова, которые можно встретить в английском языке.

Seed:

" ut it did not
appear, and after a minute or two she walked on in the
direction in
which the march ha "
re and said to herself, 'it ses all the sureo so think '

'i sooh d saie to here ' said the dat.rpillar.

'well, it soate'o toter in it wiu ' said the doymouse, ''what
so tee thet serer hereen to ae in dronint '

'ie, thay soued in the sirs,o telel,' said alice, 'and whu at
the soeen si beaan! i mase then '

'in, i saad'toa' soe kad aevee in ' she matthr said, 'and that
to tee then the was a little soreoe,'

'it i g netely toted ie a gittln oi the sias?' said the manch
hare.

'welt io a lott and that ' said the doymouse, whth a seeel af
inre the sablit horsing. ''then soen a senel sireo oi the
sirt,' said the manch hare.

'welt io a lott and that ' said the doymouse, whth a seeel af
inre the sablit horsing. ''then soen a senel sireo oi the
sirt,' said the manch hare.

'welt io a lott and that ' said the doymouse, whth a seeel af
inre the sablit horsing. ''then soen a senel sireo oi the
sirt,' said the manch hare.

'welt io a lott and that ' said the doymouse, whth a seeel af
inre the

Выше представлен текст после последней эпохи. Где-то можно увидеть небольшие фразы, но в целом текст всё ещё не понятен.

Как видно, с каждой новой эпохой выход становится менее похожим на просто набор букв и начинают прослеживаться очертания связного текста, однако многие символы не объединились в существующие слова. Для ознакомления с этой темой вполне достойный результат. Если же использовать другие способы представления исходных данных и использовать другие параметры сети, то можно добиться лучшего результата.

Вывод.

В ходе выполнения данной работы произведено ознакомление метода генерации текста с помощью рекуррентных нейронных сетей и работа с Callback в Keras.

ПРИЛОЖЕНИЕ А

ИСХОДНЫЙ КОД

```
import sys

import numpy
from tensorflow.keras import layers
from tensorflow.keras.callbacks import Callback
from tensorflow.keras.models import Sequential
from tensorflow.keras.utils import to_categorical

filename = "wonderland.txt"
raw_text = open(filename).read()
raw_text = raw_text.lower()

chars = sorted(list(set(raw_text)))
char_to_int = dict((c, i) for i, c in enumerate(chars))
int_to_char = dict((i, c) for i, c in enumerate(chars))

n_chars = len(raw_text)
n_vocab = len(chars)
print("Total Characters: ", n_chars)
print("Total Vocab: ", n_vocab)

seq_length = 100
dataX = []
dataY = []

for i in range(0, n_chars - seq_length, 1):
    seq_in = raw_text[i : i + seq_length]
    seq_out = raw_text[i + seq_length]
    dataX.append([char_to_int[char] for char in seq_in])
    dataY.append(char_to_int[seq_out])
n_patterns = len(dataX)
print("Total Patterns: ", n_patterns)

X = numpy.reshape(dataX, (n_patterns, seq_length, 1))
X = X / float(n_vocab)
y = to_categorical(dataY)

def generate_sequence(model):
    start = numpy.random.randint(0, len(dataX) - 1)
    pattern = dataX[start]
    print("Seed:")
    print('"' + "".join([int_to_char[value] for value in pattern]),
    '"')
    for i in range(1000):
        x = numpy.reshape(pattern, (1, len(pattern), 1))
        x = x / float(n_vocab)
```

```

        prediction = model.predict(x, verbose=0)
        index = numpy.argmax(prediction)
        result = int_to_char[index]
        seq_in = [int_to_char[value] for value in pattern]
        sys.stdout.write(result)
        pattern.append(index)
        pattern = pattern[1:len(pattern)]

class CustomCallback(Callback):
    def __init__(self, epochs):
        super(CustomCallback, self).__init__()
        self.epochs = epochs

    def on_epoch_end(self, epoch, logs=None):
        if epoch in self.epochs:
            generate_sequence(model)

model = Sequential()
model.add(layers.LSTM(256, input_shape=(X.shape[1], X.shape[2])))
model.add(layers.Dropout(0.2))
model.add(layers.Dense(y.shape[1], activation="softmax"))
model.compile(loss="categorical_crossentropy", optimizer="adam")

model.fit(
    X, y, epochs=20, batch_size=128, callbacks=[CustomCallback([0,
9, 14, 19])]
)

```