In [ ]:
```
import datetime
import numpy as np
import tensorflow as tf
```

```
2023-11-10 17:25:09.606327: I tensorflow/core/util/util.cc:169] oneDNN cus
tom operations are on. You may see slightly different numerical results du
e to floating-point round-off errors from different computation orders. To
turn them off, set the environment variable `TF_ENABLE_ONEDNN_OPTS=0`.
```

In [ ]:
```
import numpy as np
# (*func7*) one_hot_encode
def one_hot_encode(tags, mapping):
    # create empty vector
    encoding = np.zeros(len(mapping), dtype='uint8')
    # mark 1 for each tag in the vector
    for tag in tags:
        encoding[mapping[tag]] = 1
    return encoding
```

In [ ]:
```
# (*func6*) create_tag_mapping -> 7
# combineddata_dev = train_dev_y + test_dev_y, 'FixedByID'
def create_tag_mapping(mapping_csv, tagname):
    # tagname FixedByID
    print('tagname', tagname)
    # create a set of all known tags
    labels = set()
    IssueType_Tags = []
    for i in range(len(mapping_csv)):
        # convert spaced separated tags into an array of tags
        tags = mapping_csv[i].split('|')
        # add tags to the set of known labels
        labels.update(tags)
    # labels 存 tag(dev)
    # convert set of labels to a list to list
    labels = list(labels)
    # order set alphabetically
    labels.sort()
    # label和数字的正反映射
    # dict that maps labels to integers, and the reverse
    labels_map = {labels[i]: i for i in range(len(labels))}
    inv_labels_map = {i: labels[i] for i in range(len(labels))}
    # mapping_csv转为onehot编码
    for i in range(len(mapping_csv)):
        # Create One Hot Encoding For Issue Type
        IssueType_Tag = one_hot_encode(mapping_csv[i].split('|'), labels_
        IssueType_Tags.append(IssueType_Tag)

    result = IssueType_Tags
    return labels_map, inv_labels_map, result
```

In [ ]:
```
# (*func8*) RemoveTestRecordIfNotExistInTrainData
```

```
def RemoveTestRecordIfNotExistInTrainData(traindata, testdata):
    traingroup = traindata.groupby(["Name", "FixedByID"], as_index=True)[
    testgroup = testdata.groupby(["Name", "FixedByID"], as_index=True)["F
    for ind in testgroup.index:
        try:
            record = traindata[
                traindata['FixedByID'].str.match(testgroup['FixedByID'][i
                    testgroup['Name'][ind])]
            if len(record) < 1:
                print('remove from testdata...')
                testdata = testdata.drop(testdata[
                                                testdata['FixedByID'].str.ma
                                                    'Name'].str.match(testgr
        except:
            print("An exception occurred index :", ind)
    return testdata
```

In [ ]:
```
# Setup Project Parameters
DataAugmentation, DataAugThreshold = True, 30000
DataFilePath, DataFileName, FileType = "Data/", "IssueaspnetcoreWebScrap"
MAX_SEQUENCE_LENGTH, EMBEDDING_DIM = 300, 100
LoadDataAugFromFile = False
LearningRate = 0.001
VALIDATION_SPLIT = 0.2
```

In [ ]:
```
# log file
filename = 'Multimodel' + '_' + DataFileName + '_' + "dataaug" + '_' + st
filelog = open(filename + ".txt", "w")
filelog.write("StartTime:" + str(datetime.datetime.now()))
filelog.close()
```

In [ ]:
```
# get traindata & testdata
import pandas as pd
traindata = pd.read_csv("Data/IssueaspnetcoreWebScraptrainaugdata5.csv",
                                error_bad_lines=False, index_col=False, d
                                low_memory=False).sample(frac=1)
traindata = traindata.rename(columns={'ï»¿RepoID': 'RepoID'}, inplace=Fal
testdata = pd.read_csv("Data/IssueaspnetcoreWebScraptestdata5.csv",
                               error_bad_lines=False, index_col=False, dt
                               low_memory=False).sample(frac=1)
testdata = testdata.rename(columns={'ï»¿RepoID': 'RepoID'}, inplace=False
testdata = RemoveTestRecordIfNotExistInTrainData(traindata, testdata)
```

In [ ]:
```
# /*****************Train Data***********************/
train_dev_y = list(traindata['FixedByID'])  # Developer List
train_btype_y = list(traindata['Name'])  # Bug Type List
train_x_context = list(traindata['Title_Description'])
traindata.AST = traindata.AST.astype(str)
train_x_AST = list(traindata['AST'])

x_train_context = train_x_context
x_train_AST = train_x_AST
```

```
# /****************Test Data************************/
x_test_context = list(testdata['Title_Description'])
x_test_AST = list(testdata['Title_Description'])
test_dev_y = list(testdata['FixedByID'])  # Developer List
test_btype_y = list(testdata['Name'])  # Bug Type List
```

In [ ]:
```
# /******************Label Encoder*********************/
### Developer Encoder
combineddata_dev = train_dev_y + test_dev_y
# (*func6*) create_tag_mapping 把train_dev_y 和 test_dev_y 由str进行onehot
dev_labels_map, dev_inv_labels_map, combineddata_dev_enc = create_tag_map
dev_y_train = combineddata_dev_enc[:len(train_dev_y)]
dev_y_test = combineddata_dev_enc[len(train_dev_y):]
# 31929 1430 33359
print("Developer", "Training: ", len(train_dev_y), "Testing :", len(test_
        len(combineddata_dev_enc))

### BugType Encoder
combineddata_bugtype = train_btype_y + test_btype_y
# 同理编码bugtype
btype_labels_map, btype_inv_labels_map, combineddata_bugtype_enc = create
btype_y_train = combineddata_bugtype_enc[:len(train_btype_y)]
btype_y_test = combineddata_bugtype_enc[len(train_btype_y):]
print("Bug Type", "Training: ", len(btype_y_train), "Testing :", len(btyp
        len(combineddata_bugtype_enc))
```

```
tagname FixedByID
Developer Training:  31929 Testing : 1363 Combined DEV + TEST 33292
tagname Name
Bug Type Training:  31929 Testing : 1363 Combined DEV + TEST 33292
```

In [ ]:
```
from keras.preprocessing.text import Tokenizer
# /******************Tokenizer***************************/

## context 小写后 转为 toeknizer
x_train_context = [str(row).lower() for row in x_train_context]
x_test_context = [str(row).lower() for row in x_test_context]
combineddata_context = x_train_context + x_test_context
## 不限词汇表大小，以单词为单位，未知词标为Unknown，过滤掉'!"#$%&()*+,-./:;<=>?@[
tk_context = Tokenizer(num_words=None, char_level=None, oov_token='Unknow
                        filters='!"#$%&()*+,-./:;<=>?@[\\]^_`{|}~\t\n')
tk_context.fit_on_texts(combineddata_context)

## AST 小写后 转为 toeknizer
x_train_AST = [str(row).lower() for row in x_train_AST]
x_test_AST = [str(row).lower() for row in x_test_AST]
combineddata_AST = x_train_AST + x_test_AST
tk_AST = Tokenizer(num_words=None, char_level=None, oov_token='Unknown')
tk_AST.fit_on_texts(combineddata_AST)
```

In [ ]:
```
# Convert string to index
x_train_context_sequences = tk_context.texts_to_sequences(x_train_context
x_train_AST_sequences = tk_AST.texts_to_sequences(x_train_AST)
x_test_context_sequences = tk_context.texts_to_sequences(x_test_context)
x_test_AST_sequences = tk_AST.texts_to_sequences(x_test_AST)
```

In [ ]:
```
from tensorflow.keras.preprocessing.sequence import pad_sequences
# Padding
x_train_context = pad_sequences(x_train_context_sequences, maxlen=MAX_SEQ
x_train_AST = pad_sequences(x_train_AST_sequences, maxlen=MAX_SEQUENCE_LE
x_test_context = pad_sequences(x_test_context_sequences, maxlen=MAX_SEQUE
x_test_AST = pad_sequences(x_test_AST_sequences, maxlen=MAX_SEQUENCE_LENG

# Convert to numpy array
x_train_context = np.array(x_train_context)
x_train_AST = np.array(x_train_AST)
x_test_context = np.array(x_test_context)
x_test_AST = np.array(x_test_AST)
```

In [ ]:
```
noofbugtype = len(btype_labels_map)
noofdev = len(dev_labels_map)
btype_y_train = np.array(btype_y_train)
dev_y_train = np.array(dev_y_train)
btype_y_test = np.array(btype_y_test)
dev_y_test = np.array(dev_y_test)
```

In [ ]:
```
# Visualize Model
logdir = "logs/"
tensorboard_callback = tf.keras.callbacks.TensorBoard(log_dir=logdir)
# A model.fit() training loop will check at end of every epoch whether th
earlystop = tf.keras.callbacks.EarlyStopping(monitor='loss', patience=3)

starttime = datetime.datetime.now()
print("Start Time =", starttime)
print('Predict Developer')
```

```
Start Time = 2023-11-10 17:45:32.508875
Predict Developer
```

In [ ]:
```
from keras.layers import Input
# inputs
input_context = Input(shape=(MAX_SEQUENCE_LENGTH,), dtype=tf.float32,
                        name="Bug_TitleandDescription")  # Bug Title and
input_AST = Input(shape=(MAX_SEQUENCE_LENGTH,), dtype=tf.float32, name="B
```

In [ ]:
```
from keras.layers import Embedding, Conv1D, GlobalMaxPool1D, Flatten
# Context Enconder
emb_Context = Embedding(input_dim=len(tk_context.word_index) + 2, input_l
                        output_dim=EMBEDDING_DIM, name="Context_Embedding
conv_Context = Conv1D(filters=64, kernel_size=2, padding='same', activati
                        name="Context_Convolutional_Layer")(emb_Context)
maxpool_Context = GlobalMaxPool1D(name="Context_Maxpool_Layer")(conv_Cont
flatcon = Flatten(name="Context_Flatten_Layer")(maxpool_Context)
2023-11-10 17:48:29.620237: E tensorflow/stream_executor/cuda/cuda_driver.
cc:271] failed call to cuInit: CUDA_ERROR_NO_DEVICE: no CUDA-capable devic
e is detected
2023-11-10 17:48:29.620264: I tensorflow/stream_executor/cuda/cuda_diagnos
tics.cc:163] no NVIDIA GPU device is present: /dev/nvidia0 does not exist
2023-11-10 17:48:29.620995: I tensorflow/core/platform/cpu_feature_guard.c
c:193] This TensorFlow binary is optimized with oneAPI Deep Neural Network
Library (oneDNN) to use the following CPU instructions in performance-crit
ical operations:  AVX2 AVX512F AVX512_VNNI AVX512_BF16 FMA
To enable them in other operations, rebuild TensorFlow with the appropriat
e compiler flags.
```

In [ ]:

```
from keras.layers import Embedding, Bidirectional, LSTM, GlobalMaxPool1D,
# AST Enconder
emb_AST = Embedding(input_dim=len(tk_AST.word_index) + 2, input_length=MA
                    output_dim=EMBEDDING_DIM, name="AST_Embedding")(input
bilstm_AST = Bidirectional(LSTM(25, return_sequences=True, name="AST_LSTM
maxpool_AST = GlobalMaxPool1D(name="AST_Maxpool_Layer")(bilstm_AST)
flatAST = Flatten(name="AST_Flatten_Layer")(maxpool_AST)
```

In [ ]:

```
from keras.layers import concatenate, BatchNormalization, Dropout, Dense
from keras.models import Model
cat = concatenate([flatcon, flatAST], name="Concatenate_Flatten_Layer")

bn = BatchNormalization()(cat)
drop = Dropout(0.5)(bn)
dense = Dense(50, activation='relu')(drop)
DevOutput = Dense(noofdev, activation='sigmoid', name="Developer")(dense)
BugTypeOutput = Dense(noofbugtype, activation='sigmoid', name="Bug_Type")
Bil_LSTM_MultiTask_model = Model(inputs=[input_context, input_AST], outpu
```

In [ ]:

```
from tensorflow.keras.optimizers import Adam
from evaluation import *
Bil_LSTM_MultiTask_model.compile(loss='binary_crossentropy', optimizer=Ad
                                 metrics=['accuracy', c_precision, c_recal
                                          hammingloss])

history = Bil_LSTM_MultiTask_model.fit([x_train_context, x_train_AST],
                                        [dev_y_train, btype_y_train],
                                        callbacks=[tensorboard_callback,
                                        epochs=50, verbose=2, validation_
```

```
Epoch 1/50
```

In [ ]:

```
from keras.callbacks import CSVLogger
csv_logger = CSVLogger(filename + '.csv', append=True, separator=';')
Bil_LSTM_MultiTask_model.evaluate([x_test_context, x_test_AST], [dev_y_te

filelog = open(filename + ".txt", "a")
filelog.write("Endtime:" + str(datetime.now()))
filelog.close()
2023-11-10 17:57:34.620738: I tensorflow/core/util/util.cc:169] oneDNN cus
tom operations are on. You may see slightly different numerical results du
e to floating-point round-off errors from different computation orders. To
turn them off, set the environment variable `TF_ENABLE_ONEDNN_OPTS=0`.
---------------------------------------------------------------------------
-
NameError                                 Traceback (most recent call las
t)
Input In [1], in <cell line: 2>()
      1 from keras.callbacks import CSVLogger
----> 2 csv_logger = CSVLogger(filename + '.csv', append=True, separator
=';')
      3 Bil_LSTM_MultiTask_model.evaluate([x_test_context, x_test_AST], [d
ev_y_test, btype_y_test], callbacks=[csv_logger])
      5 filelog = open(filename + ".txt", "a")

NameError: name 'filename' is not defined
```

```
In [ ]:
```