

▼ Sentiment Analysis using GRU

Importing the required libraries

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt

from keras.datasets import imdb
from keras.utils import pad_sequences
from keras.models import Sequential
from keras.layers import GRU, Embedding, Dense, Dropout
```

Set hyperparameters

```
MAX_FEATURES = 5000 # number of words to keep from the dataset
MAXLEN = 250 # maximum length of a sequence
BATCH_SIZE = 32
EPOCHS = 3
```

Load the IMDB dataset

```
(X_train, y_train), (X_test, y_test) = imdb.load_data(num_words=MAX_FEATURES)

Downloading data from https://storage.googleapis.com/tensorflow/tf-keras-datasets/imdb.npz
17464789/17464789 [=====] - 0s 0us/step
```

Pad sequences to the same length

```
X_train = pad_sequences(X_train, maxlen=MAXLEN)
X_test = pad_sequences(X_test, maxlen=MAXLEN)
```

Define the model

```
model = Sequential()
model.add(Embedding(input_dim=MAX_FEATURES, output_dim=128, input_length=MAXLEN))
model.add(GRU(units=64, dropout=0.2, recurrent_dropout=0.2))
model.add(Dense(1, activation='sigmoid'))
```

Saving...

will not use cuDNN kernels since it doesn't meet the criteria. It will use a generic GPU kernel as fal

Compile the model

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

Train the model

```
history = model.fit(X_train, y_train, batch_size=BATCH_SIZE, epochs=EPOCHS, validation_data=(X_test, y_test))

Epoch 1/3
782/782 [=====] - 841s 1s/step - loss: 0.4224 - accuracy: 0.8022 - val_loss: 0.3496 - val_accuracy: 0.8500
Epoch 2/3
782/782 [=====] - 799s 1s/step - loss: 0.2701 - accuracy: 0.8921 - val_loss: 0.2849 - val_accuracy: 0.8841
Epoch 3/3
782/782 [=====] - ETA: 0s - loss: 0.1955 - accuracy: 0.9254
```

Evaluate the model on test data

```
score, acc = model.evaluate(X_test, y_test, batch_size=BATCH_SIZE)
print('Test score:', score)
print('Test accuracy:', acc)

782/782 [=====] - 54s 70ms/step - loss: 0.2883 - accuracy: 0.8837
Test score: 0.2882551848888397
Test accuracy: 0.8837199807167053
```