In [2]:
```
import jsonlines

def read_jsonl(file_path):
    sentence_id = []; summary = []; sentence = []
    with jsonlines.open(file_path, 'r') as reader:
        for line in reader:
            sentence_id.append(line['id'])
            summary.append(line['summaries'][0])
            sentence.append(line['text'])
    return sentence_id,summary,sentence

file_path = 'rl-sentence-compression\data\test-data\new
s_id,summary,sentence = read_jsonl(file_path)
```

In [12]:
```
columns = ['s_id','sentence']
data = df[columns]
data.head()
```

Out[12]:

	s_id	sentence
0	newsroom-val-title-0	Real Madrid have confirmed they have agreed to...
1	newsroom-val-title-1	American Pie singer Don McLean was arrested on...
2	newsroom-val-title-2	A candidate for governor of the northern Mexic...
3	newsroom-val-title-3	Bill Parcells, the two-time Super Bowl-winning...
4	newsroom-val-title-4	IBM’s data crunching service for the healthcar...

In [13]:
```
import nltk
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
from nltk.stem import PorterStemmer
import string

nltk.download('punkt')
nltk.download('stopwords')

stemmer = PorterStemmer()
stop_words = set(stopwords.words('english'))

def preprocess_text(text):
    text = text.lower()
    tokens = word_tokenize(text)
    processed_text = ' '.join(tokens)
    return processed_text

data['processed_sentence'] = data['sentence'].apply(pre
data.head()
```

```
[nltk_data] Downloading package punkt to /root/nltk_da
ta...
[nltk_data]   Package punkt is already up-to-date!
[nltk_data] Downloading package stopwords to /root/nlt
k_data...
[nltk_data]   Package stopwords is already up-to-date!
```

Out[13]:

	s_id	sentence	processed_sentence
0	newsroom-val-title-0	Real Madrid have confirmed they have agreed to...	real madrid confirm agre sign mexican striker ...
1	newsroom-val-title-1	American Pie singer Don McLean was arrested on...	american pie singer mclean arrest misdemeanor ...
2	newsroom-val-title-2	A candidate for governor of the northern Mexic...	candid governor northern mexican state tamauli...
3	newsroom-val-title-3	Bill Parcells, the two-time Super Bowl-winning...	bill parcel two-tim super bowl-win coach rejoi...
4	newsroom-val-title-4	IBM’s data crunching service for the healthcar...	ibm ’ data crunch servic healthcar industri wa...

In [14]:
```
print(data['sentence'][0])
```

```
Real Madrid have confirmed they have agreed to sign th
e Mexican striker Javier Hernández on a season-long lo
an from Manchester United.
```

In [15]:
```
print(data['processed_sentence'][0])
```

```
real madrid confirm agre sign mexican striker javier h
ernández season-long loan manchest unit
```

In [16]:
```
import numpy as np

def load_glove_embeddings(file_path):
    embeddings_index = {}
    with open(file_path, 'r', encoding='utf-8') as f:
        for line in f:
            values = line.split()

            embeddings_index[word] = coefs
    return embeddings_index

word_embeddings = load_glove_embeddings(glove_path)
embedding_dim = len(next(iter(word_embeddings.values())))
print("Embedding Dimension:", embedding_dim)
```

```
Embedding Dimension: 100
```

In [17]:
```
print(word_embeddings['the'])
```

```
[-0.038194 -0.24487   0.72812  -0.39961   0.083172  0.
043953 -0.39141
  0.3344   -0.57545   0.087459  0.28787  -0.06731   0.
30906  -0.26384
 -0.13231  -0.20757   0.33395  -0.33848  -0.31743  -0.
48336   0.1464
 -0.37304   0.34577   0.052041  0.44946  -0.46971   0.
02628  -0.54155
 -0.15518  -0.14107  -0.039722  0.28277   0.14393   0.
23464  -0.31021
  0.086173  0.20397   0.52624   0.17164  -0.082378 -0.
71787  -0.41531
  0.20335  -0.12763   0.41367   0.55187   0.57908  -0.
33477  -0.36559
 -0.54857  -0.062892  0.26584   0.30205   0.99775  -0.
80481  -3.0243
  0.01254  -0.36942   2.2167    0.72201  -0.24978   0.
92136   0.034514
  0.46745   1.1079   -0.19358  -0.074575  0.23353  -0.
052062 -0.22044
  0.057162 -0.15806  -0.30798  -0.41625   0.37972   0.
15006  -0.53212
 -0.2055   -1.2526    0.071624  0.70565   0.49744  -0.
42063   0.26148
 -1.538    -0.30223  -0.073438 -0.28312   0.37104  -0.
25217   0.016215
 -0.017099 -0.38984   0.87424  -0.72569  -0.51058  -0.
52028  -0.1459
  0.8278    0.27062 ]
```

In [18]:

```
import numpy as np
from tensorflow.keras.preprocessing.sequence import pad
from gensim.models import KeyedVectors

word_vectors = word_embeddings

def sentence_to_embeddings(sentence, word_vectors, embe
    words = sentence.split()
    embeddings = []
    for word in words:
        if word in word_vectors:
            embeddings.append(word_vectors[word])
        else:
            embeddings.append(np.zeros(embedding_dim))
    return embeddings

data.head()
```

Out[18]:

	s_id	sentence	processed_sentence	embeddings	padded
0	newsroom-val-title-0	Real Madrid have confirmed they have agreed to...	real madrid confirm agre sign mexican striker ...	[[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,...	[[0, 0, 0, 0,
1	newsroom-val-title-1	American Pie singer Don McLean was arrested on...	american pie singer mclean arrest misdemeanor ...	[[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,...	[[0, 0, 0, 0,
2	newsroom-val-title-2	A candidate for governor of the northern Mexic...	candid governor northern mexican state tamauli...	[[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,...	[[0, 0, 0, 0,
3	newsroom-val-title-3	Bill Parcells, the two-time Super Bowl-winning...	bill parcel two-tim super bowl-win coach rejoi...	[[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,...	[[0, 0, 0, 0,
4	newsroom-val-title-4	IBM’s data crunching service for the healthcar...	ibm ’ data crunch servic healthcar industri wa...	[[0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0, 0.0,...	[[0, 0, 0, 0,

In [28]:

```
temp = df['summary']
temp = pd.DataFrame(temp)
temp.head()
```

Out[28]:

	summary
0	Real Madrid sign Javier Hernández on loan from...
1	American Pie singer Don Mclean arrested on dom...
2	Candidate for governor of Mexican state of Tam...
3	Bill Parcells rejoining ESPN for third time
4	IBM Watson Health now counts CVS Health as a p...

In [33]:

```
word_vectors = word_embeddings

def sentence_to_embeddings(sentence, word_vectors, embe
    words = sentence.split()
    embeddings = []
    for word in words:
        if word in word_vectors:
            embeddings.append(word_vectors[word])
        else:
            embeddings.append(np.zeros(embedding_dim))
    return embeddings

max_seq_length = 50
temp['s_padded_embeddings'] = pad_sequences(temp['s_emb

temp.head()
```

Out[33]:

	summary	processed_summary	s_embeddings	s_padded_embed
0	Real Madrid sign Javier Hernández on loan from...	real madrid sign javier hernández loan manches...	[[0.45006, 0.15098, 0.31014, -0.20369, -0.2210...	[[0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
1	American Pie singer Don Mclean arrested on dom...	american pie singer mclean arrest domest viole...	[[0.38666, 0.64827, 0.72807, -0.077056, 0.1545...	[[0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
2	Candidate for governor of Mexican state of Tam...	candid governor mexican state tamaulipa kill s...	[[-0.33871, -0.37143, 0.4443, 0.72357, -0.3119...	[[0, 0, 0, 0, 0, 0, 0, 0, 0, 0,
3	Bill Parcells rejoining ESPN for third time	bill parcel rejoin espn third time	[[-0.10535, -0.025048, 0.55525, -1.0371, 0.221...	[[0, 0, 0, -1, 0, 0, 0, 0, 0, 0,
4	IBM Watson Health now counts CVS Health as a p...	ibm watson health count cv health partner	[[0.4875, 0.4214, 0.013491, 0.71504, 0.3708, -...	[[0, 0, 0, 0, 0, 0, 0, 0, 0, 0,

In [35]:
```
final.columns
```

Out[35]:
```
Index(['s_id', 'sentence', 'processed_sentence', 'embeddings',
       'padded_embeddings', 'summary', 'processed_summary', 's_embeddings',
       's_padded_embeddings'],
      dtype='object')
```

In [121]:

```
import numpy as np
from tensorflow.keras.layers import Input, Embedding, L

max_words = 10000
max_seq_len = 100

tokenizer = Tokenizer(num_words=max_words)
tokenizer.fit_on_texts(final['processed_sentence'])
tokenizer.fit_on_texts(final['processed_summary'])

sentence_sequences = pad_sequences(sentence_sequences,
summary_sequences = pad_sequences(summary_sequences, ma

embedding_dim = 100
hidden_units = 128

embedding_layer = Embedding(input_dim=max_words, output

sentence_embedding = embedding_layer(sentence_input)
summary_embedding = embedding_layer(summary_input)

sentence_rnn = lstm_layer(sentence_embedding)
summary_rnn = lstm_layer(summary_embedding)
```

In [122]:
```
from tensorflow.keras.utils import to_categorical

model = Model(inputs=[sentence_input, summary_input], o
model.compile(optimizer='adam', loss='categorical_cross

summary_sequences_one_hot = to_categorical(summary_sequ

model.fit([sentence_sequences, summary_sequences], summ
```

```
Epoch 4/10
7/7 [==============================] - 7s 1s/step
- loss: 4.3803 - accuracy: 0.9211 - val_loss: 3.07
68 - val_accuracy: 0.9184
Epoch 5/10
7/7 [==============================] - 6s 950ms/st
ep - loss: 2.2278 - accuracy: 0.9211 - val_loss:
1.3990 - val_accuracy: 0.9184
Epoch 6/10
7/7 [==============================] - 7s 1s/step
- loss: 1.0837 - accuracy: 0.9211 - val_loss: 0.90
76 - val_accuracy: 0.9184
Epoch 7/10
7/7 [==============================] - 6s 947ms/st
ep - loss: 0.8279 - accuracy: 0.9211 - val_loss:
0.8369 - val_accuracy: 0.9184
Epoch 8/10
7/7 [==============================] - 7s 1s/step
- loss: 0.7835 - accuracy: 0.9211 - val_loss: 0.82
46 - val_accuracy: 0.9184
```

In [143]:

```
from nltk.translate.bleu_score import sentence_bleu
from nltk.translate.bleu_score import SmoothingFunction
from rouge import Rouge
from tensorflow.keras.preprocessing.sequence import pad

def preprocess_input_and_summary(input_sentence, expect
    processed_input_sentence = preprocess_text(input_se
    processed_expected_summary = preprocess_text(expect
    return processed_input_sentence, processed_expected

def decode_summary(summary_sequence, tokenizer):
    decoded_summary = tokenizer.sequences_to_texts(summ
    decoded_summary = [sentence.split() for sentence in
    decoded_summary = [' '.join(sentence) for sentence
    return decoded_summary[0]

def compress(input_sentence,expected_summary):
    processed_input_sentence, processed_expected_summar

    bleu_score = sentence_bleu([processed_expected_summ
    rouge = Rouge()
    rouge_scores = rouge.get_scores(decoded_summary, pr

    print("Summary:", decoded_summary)
    print("BLEU Score:", bleu_score)
```