In [1]:

```python
from google.colab import drive
drive.mount('/content/drive')
```

```
Mounted at /content/drive
```

In [2]:

```python
import os
import pathlib
from pathlib import Path
os.chdir("/content/drive/My Drive/Akarshan/BERT")
!ls -l
```

```
total 51436
-rw------- 1 root root  8388432 Dec 26 21:48  BERT5.hdf5
drwx------ 2 root root     4096 Dec  3 16:27  clr
-rw------- 1 root root   488019 Dec 26 22:59  Compare.ipynb
-rw------- 1 root root   459928 Dec 26 23:02 'Copy of EDA on results.ipynb'
drwx------ 2 root root     4096 Dec  3 16:27  Data
-rw------- 1 root root  8306584 Dec 24 07:57  DBert1hk.hdf5
-rw------- 1 root root 12719136 Dec 24 07:57  DBert4hk.hdf5
-rw------- 1 root root   251029 Dec 26 22:52  Distllbert400000.ipynb
-rw------- 1 root root   476324 Dec 26 22:54 'EDA on results.ipynb'
drwx------ 2 root root     4096 Dec 18 07:14 'misc model'
-rw------- 1 root root    42964 Dec 26 22:44  model.png
drwx------ 2 root root     4096 Dec  3 16:27  papers
-rw------- 1 root root  8306584 Dec 19 08:56  Rbert4.hdf5
-rw------- 1 root root   203164 Dec 26 21:29  Retraining.ipynb
-rw------- 1 root root    85578 Dec 26 22:54  Roberta.ipynb
-rw------- 1 root root 12719160 Dec 25 10:35  SBert.hdf5
-rw------- 1 root root   203468 Dec 26 22:53  SciBert400k.ipynb
```

In [3]:

```python
!pip install transformers
!pip install pympler
!pip install tensorflow_addons
```

```
Collecting transformers
  Downloading transformers-4.15.0-py3-none-any.whl (3.4 MB)
     |████████████████████████████████| 3.4 MB 5.3 MB/s
Collecting tokenizers<0.11,>=0.10.1
  Downloading tokenizers-0.10.3-cp37-cp37m-manylinux_2_5_x86_64.manylinux1_x86_64.manylin
ux_2_12_x86_64.manylinux2010_x86_64.whl (3.3 MB)
     |████████████████████████████████| 3.3 MB 44.3 MB/s
Requirement already satisfied: numpy>=1.17 in /usr/local/lib/python3.7/dist-packages (fro
m transformers) (1.19.5)
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.7/dist-packages
(from transformers) (21.3)
Requirement already satisfied: importlib-metadata in /usr/local/lib/python3.7/dist-packag
es (from transformers) (4.8.2)
Collecting sacremoses
  Downloading sacremoses-0.0.46-py3-none-any.whl (895 kB)
     |████████████████████████████████| 895 kB 38.7 MB/s
Requirement already satisfied: filelock in /usr/local/lib/python3.7/dist-packages (from t
ransformers) (3.4.0)
Requirement already satisfied: tqdm>=4.27 in /usr/local/lib/python3.7/dist-packages (from
transformers) (4.62.3)
Requirement already satisfied: regex!=2019.12.17 in /usr/local/lib/python3.7/dist-package
s (from transformers) (2019.12.20)
Requirement already satisfied: requests in /usr/local/lib/python3.7/dist-packages (from t
ransformers) (2.23.0)
Collecting pyyaml>=5.1
  Downloading PyYAML-6.0-cp37-cp37m-manylinux_2_5_x86_64.manylinux1_x86_64.manylinux_2_12
_x86_64.manylinux2010_x86_64.whl (596 kB)
     |████████████████████████████████| 596 kB 45.8 MB/s
Collecting huggingface-hub<1.0,>=0.1.0
```

```
  Downloading huggingface_hub-0.2.1-py3-none-any.whl (61 kB)
     |████████████████████████████████| 61 kB 443 kB/s
Requirement already satisfied: typing-extensions>=3.7.4.3 in /usr/local/lib/python3.7/dist-packages (from huggingface-hub<1.0,>=0.1.0->transformers) (3.10.0.2)
Requirement already satisfied: pyparsing!=3.0.5,>=2.0.2 in /usr/local/lib/python3.7/dist-packages (from packaging>=20.0->transformers) (3.0.6)
Requirement already satisfied: zipp>=0.5 in /usr/local/lib/python3.7/dist-packages (from importlib-metadata->transformers) (3.6.0)
Requirement already satisfied: urllib3!=1.25.0,!=1.25.1,<1.26,>=1.21.1 in /usr/local/lib/python3.7/dist-packages (from requests->transformers) (1.24.3)
Requirement already satisfied: chardet<4,>=3.0.2 in /usr/local/lib/python3.7/dist-packages (from requests->transformers) (3.0.4)
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.7/dist-packages (from requests->transformers) (2021.10.8)
Requirement already satisfied: idna<3,>=2.5 in /usr/local/lib/python3.7/dist-packages (from requests->transformers) (2.10)
Requirement already satisfied: six in /usr/local/lib/python3.7/dist-packages (from sacremoses->transformers) (1.15.0)
Requirement already satisfied: joblib in /usr/local/lib/python3.7/dist-packages (from sacremoses->transformers) (1.1.0)
Requirement already satisfied: click in /usr/local/lib/python3.7/dist-packages (from sacremoses->transformers) (7.1.2)
Installing collected packages: pyyaml, tokenizers, sacremoses, huggingface-hub, transformers
  Attempting uninstall: pyyaml
    Found existing installation: PyYAML 3.13
    Uninstalling PyYAML-3.13:
      Successfully uninstalled PyYAML-3.13
Successfully installed huggingface-hub-0.2.1 pyyaml-6.0 sacremoses-0.0.46 tokenizers-0.10.3 transformers-4.15.0
Collecting pympler
  Downloading Pympler-1.0.1-py3-none-any.whl (164 kB)
     |████████████████████████████████| 164 kB 5.1 MB/s
Installing collected packages: pympler
Successfully installed pympler-1.0.1
Collecting tensorflow_addons
  Downloading tensorflow_addons-0.15.0-cp37-cp37m-manylinux_2_12_x86_64.manylinux2010_x86_64.whl (1.1 MB)
     |████████████████████████████████| 1.1 MB 5.0 MB/s
Requirement already satisfied: typeguard>=2.7 in /usr/local/lib/python3.7/dist-packages (from tensorflow_addons) (2.7.1)
Installing collected packages: tensorflow-addons
Successfully installed tensorflow-addons-0.15.0
```

In [4]:

```python
import numpy as np
import pickle
import pandas as pd
import pickle
import time
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.metrics import roc_curve,auc,confusion_matrix,accuracy_score,precision_score,recall_score,f1_score
from pympler import asizeof
import tensorflow as tf
from sklearn.model_selection import train_test_split
from sklearn.metrics import classification_report
import transformers
from transformers import pipeline
from tensorflow.keras.layers import concatenate
from transformers import TFAutoModel, AutoTokenizer, AutoConfig,TFAutoModelForSequenceClassification
from tensorflow.keras.callbacks import ModelCheckpoint
from clr import clr_callback
import tensorflow_addons as tfa
```

In [5]:

```python
csvfile = 'Data//data.csv'
```

```
dropna = 'Data//datadropna.csv'
sent_data_file = 'Data//sent_data.csv'
label_file = 'Data//label.csv'
vocab_file = 'Data//vocab_tr_w.txt'
```

In [6]:

```
df = pd.read_csv(dropna,usecols = ['SBE','Label'])
# df.dropna(inplace=True)
print(df.head())
print(df.shape)
```

```
   Label                                                SBE
0      1  To facilitate an easier notation throughout th...
1      0  Therefore _MATH_ defines a special order of ti...
2      0  This is important since only _MATH_ is the rea...
3      0  Note that in all contour time-integrals we ess...
4      0  Theorem _REF_ proves the equivalence of ensemb...
(1189321, 2)
```

Working with distillbert400k as it performed the best

In [7]:

```
num = len(os.listdir('Data//embeddingBr//'))

with open('Data//embeddingBr//embeddings'+str(0),'rb') as f:
    dataD = pickle.load(f)

for idx in range(1,num):

  with open('Data//embeddingBr//embeddings'+str(idx),'rb') as f:
    mat = pickle.load(f)
    dataD=np.concatenate([dataD,mat],axis=0)
```

In [8]:

```
np.shape(dataD)
```

Out[8]:

```
(400000, 768)
```

In [9]:

```
df =  df.iloc[:np.shape(dataD)[0],:]
```

Not shuffling data for further analysis.

In [10]:

```
_, temp_text, _, temp_labels = train_test_split(dataD, df['Label'],
                                                                    random_state=2018,
                                                                    test_size=0.3,
                                                                    stratify=df['Label'
])

# we will use temp_text and temp_labels to create validation and test set
_, test_text, _, test_labels = train_test_split(temp_text, temp_labels,
                                                                random_state=2018,
                                                                test_size=0.5,
                                                                stratify=temp_labels)



test_labels = tf.keras.utils.to_categorical(test_labels)

test_data = tf.data.Dataset.from_tensor_slices((test_text))
test_data = test_data.batch(128)#no shuffle
```

In [11]:

```
from keras.models import load_model
model = load_model("BERT5.hdf5")
```

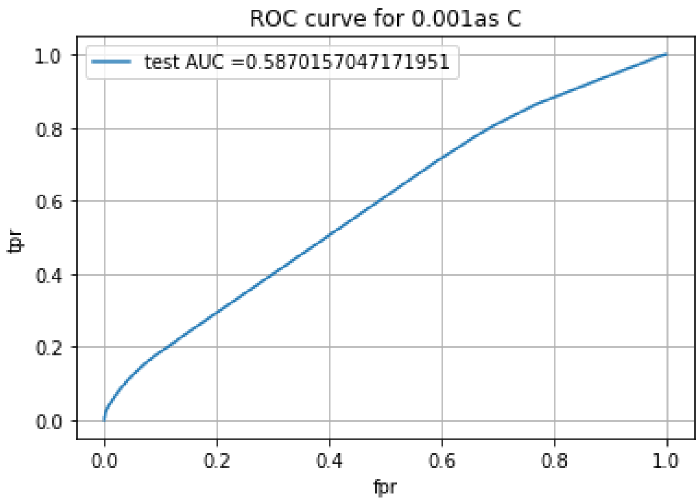
In [12]:

```
y_pr_ts_all = model.predict(test_data)
```

In [13]:

```
y_pr_ts = y_pr_ts_all[:,0]
y_ts = test_labels[:,0]
from sklearn.metrics import roc_curve,auc,confusion_matrix,accuracy_score,precision_score
,recall_score,f1_score

test_fpr, test_tpr, te_thresholds = roc_curve(y_ts, y_pr_ts)
plt.plot(test_fpr, test_tpr, label="test AUC ="+str(auc(test_fpr, test_tpr)))
plt.xlabel("fpr")
plt.ylabel("tpr")
plt.title('ROC curve for '+str (0.001)+'as C')
plt.legend()
plt.grid()
plt.show()
```



In [14]:

```
# This section of code where ever implemented is taken from sample kNN python notebook

def find_best_threshold(threshould, fpr, tpr):
    t = threshould[np.argmax(tpr*(1-fpr))]
    # (tpr*(1-fpr)) will be maximum if your fpr is very low and tpr is very high
    print("the maximum value of tpr*(1-fpr)", max(tpr*(1-fpr)), "for threshold", np.round
(t,3))
    return t

def predict_with_best_t(proba, threshould):
    predictions = []
    for i in proba:
        if i>=threshould:
            predictions.append(1)
        else:
            predictions.append(0)
    return predictions


print('test')
best_ts_thres = find_best_threshold(te_thresholds, test_fpr, test_tpr)
```

```
test
the maximum value of tpr*(1-fpr) 0.2901352933913039 for threshold 0.72
```

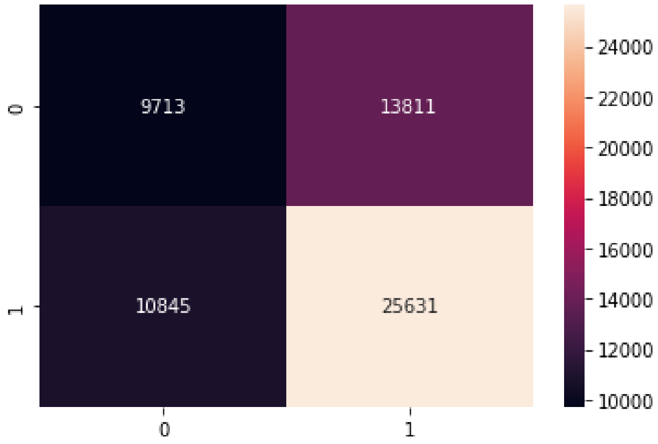
In [15]:

```
print('Test Confusion Matrix')
cm2 = pd.DataFrame(confusion_matrix(y_ts, predict_with_best_t(y_pr_ts, best_ts_thres)), range(2),range(2))
sns.heatmap(cm2, annot=True,fmt='g')
```

```
Test Confusion Matrix
```

Out[15]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f44f00571d0>
```



In [16]:

```
acc=accuracy_score(y_ts, predict_with_best_t(y_pr_ts, best_ts_thres))*100
ps=precision_score(y_ts, predict_with_best_t(y_pr_ts, best_ts_thres))*100
rc=recall_score(y_ts, predict_with_best_t(y_pr_ts, best_ts_thres))*100
f1=f1_score(y_ts, predict_with_best_t(y_pr_ts, best_ts_thres))*100

print("Accuracy on test set: %0.2f%%"%(acc))
print("Precision on test set: %0.2f%%"%(ps))
print("recall score on test set: %0.2f%%"%(rc))
print("f1 score on test set: %0.2f%%"%(f1))
```

```
Accuracy on test set: 58.91%
Precision on test set: 64.98%
recall score on test set: 70.27%
f1 score on test set: 67.52%
```

Clarification on FP and FN

1. **After the step *test_labels = tf.keras.utils.to_categorical(test_labels)*, Test_labels[0] stores values as follows: 0 for 'Editing Needed' 1 for 'Editing Not Needed'.**
2. **So TP are values which did not need editing and got classified as that(1,1).**
3. **TN are values which needed editing and got classified as that(0,0).**
4. **FP are values which needed editing but got classified otherwise(0,1).**
5. **FN are values which did not need editing but got classified otherwise(1,0).**

In [17]:

```
# with open('Data//y_pr_ts','wb') as f:
#   pickle.dump(y_pr_ts,f)
with open('Data//y_pr_ts','rb') as f:
  y_pr_ts = pickle.load(f)
# with open('Data//Preds','wb') as f:
#   pickle.dump(predict_with_best_t(y_pr_ts, best_ts_thres),f)
with open('Data//Preds','rb') as f:
  preds = pickle.load(f)
# with open('Data//y_ts','wb') as f:
#   pickle.dump(y_ts,f)
with open('Data//y_ts','rb') as f:
  y_ts = pickle.load(f)
```

In [18]:

```
df1 = pd.read_csv('Data//data2.csv')
```

```
df1 = pd.read_csv('Data//data2.csv')
print(df1.shape)
df1.dropna(subset=['SBE', 'Label'],inplace=True)
print(df1.shape)
df1 = df1.iloc[:400000,:]
df1.columns
```

```
(1189412, 7)
(1189321, 7)
```

Out[18]:

```
Index(['SID', 'Domain', 'SBE', 'SAE', 'del_word', 'ins_word', 'Label'], dtype='object')
```

In [19]:

```
df1.shape
```

Out[19]:

```
(400000, 7)
```

In [21]:

```
_, temp_text, _, temp_labels = train_test_split(df1[['Domain', 'SBE', 'SAE', 'del_word',
'ins_word']], df1['Label'],
                                                                    random_state=2018,
                                                                    test_size=0.3,
                                                                    stratify=df1['Label
'])

# we will use temp_text and temp_labels to create validation and test set
_, test_text1, _, test_labels1 = train_test_split(temp_text, temp_labels,
                                                                random_state=2018,
                                                                test_size=0.5,
                                                                stratify=temp_labels)
test_labels = tf.keras.utils.to_categorical(test_labels1)
```

Analysing False Positive

In [22]:

```
index = []
for i,(l,p) in enumerate(zip(y_ts,preds)):
  if l == 0 and p ==1:
    index.append(i)
```

In [23]:

```
cm = pd.DataFrame(confusion_matrix(y_ts, preds), range(2),range(2))
cm
```

Out[23]:

	0	1
0	9713	13811
1	10845	25631

In [24]:

```
cm.iloc[0,1]==len(index)
```

Out[24]:

```
True
```

In [25]:

```
fp = test_text1.iloc[index]
```

In [26]:

```
fplabels = np.take(test_labelsf[:,0],index)
```

In [27]:

```
np.unique(fplabels)
```

Out[27]:

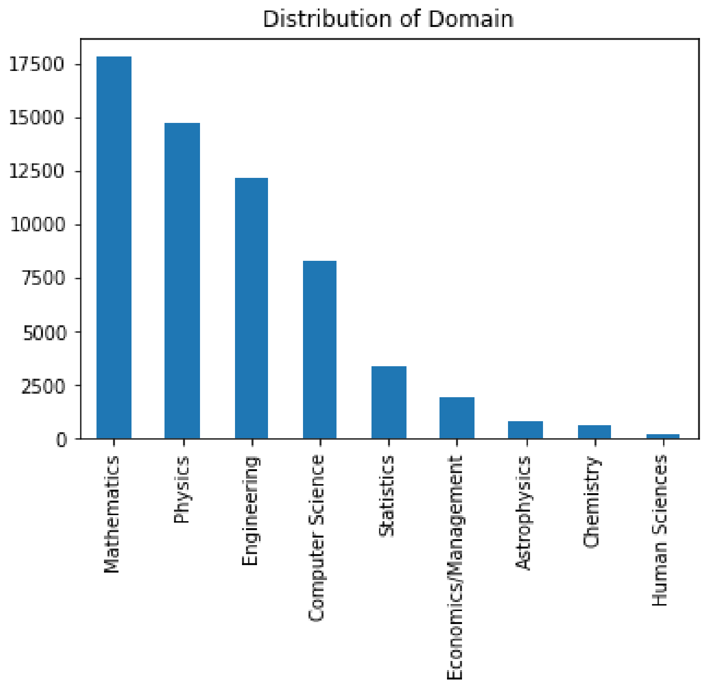
```
array([0.], dtype=float32)
```

In [28]:

```
plt.title('Distribution of Domain')
test_text1['Domain'].value_counts().plot(kind='bar')
```

Out[28]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f442930e310>
```

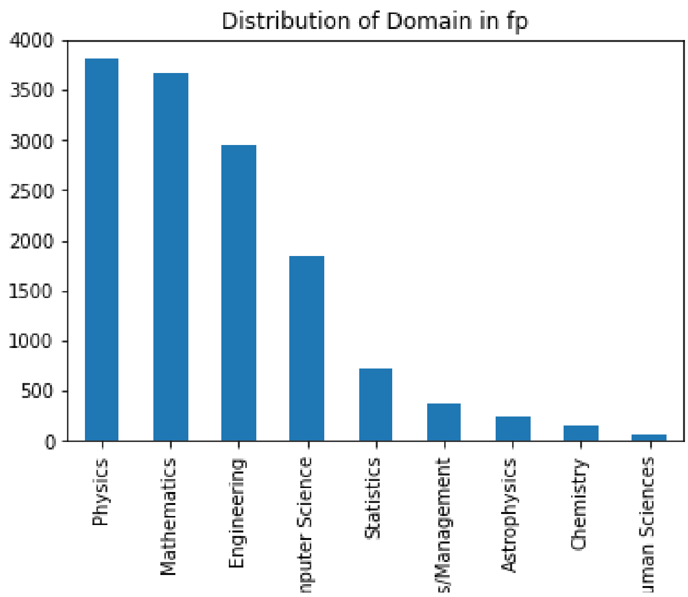


In [29]:

```
plt.title('Distribution of Domain in fp')
fp['Domain'].value_counts().plot(kind='bar')
```

Out[29]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f4429252410>
```



In [30]:

```
del_ins_pair = fp['del_word']+' '+fp['ins_word']
```

In [31]:

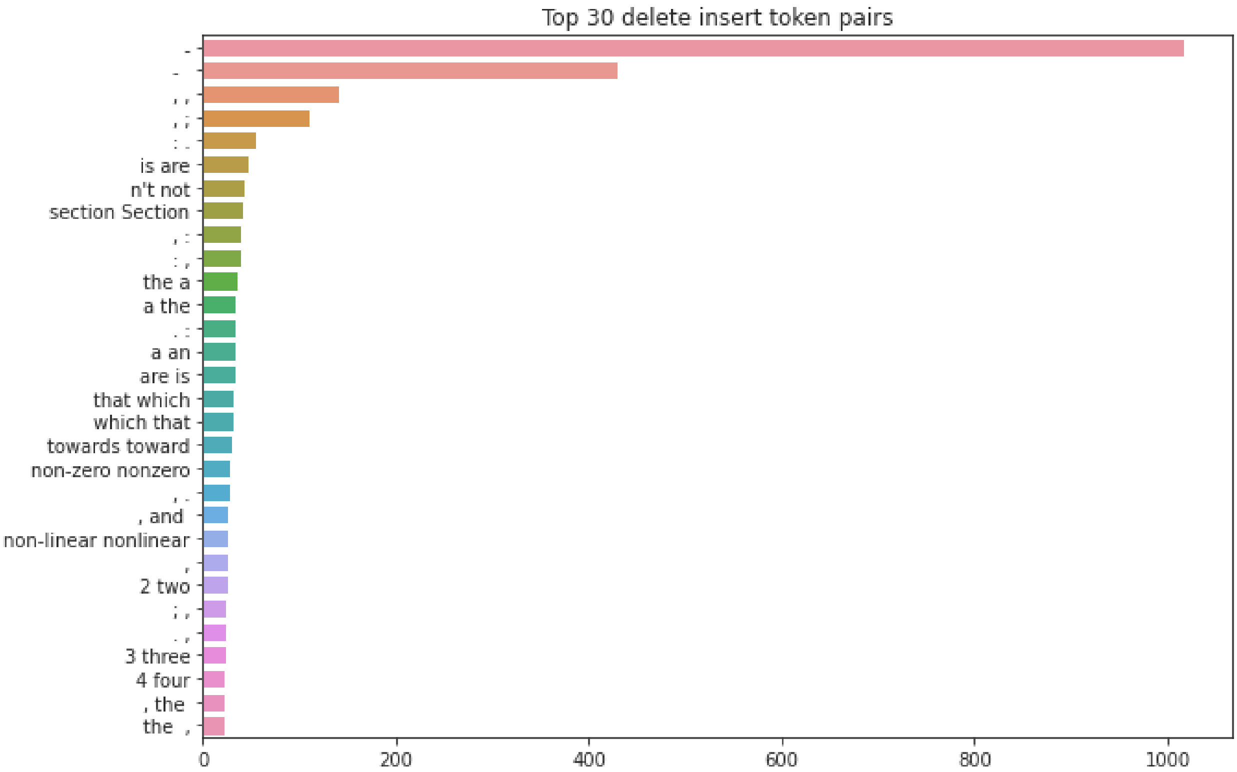
```
del_ins_pair_plt = del_ins_pair.value_counts().sort_values(ascending = False).head(30)
```

In [32]:

```
plt.figure(figsize=(10,7))
sns.set_style('ticks')
plt.title('Top 30 delete insert token pairs')
sns.barplot(y=del_ins_pair_plt.index, x= del_ins_pair_plt.values )
```

Out[32]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f44291e4550>
```



Analyzing FN

In [33]:

```
index = []
for i,(l,p) in enumerate(zip(y_ts,preds)):
  if l == 1 and p ==0:
    index.append(i)
```

In [34]:

```
cm.iloc[1,0]==len(index)
```

Out[34]:

```
True
```

In [35]:

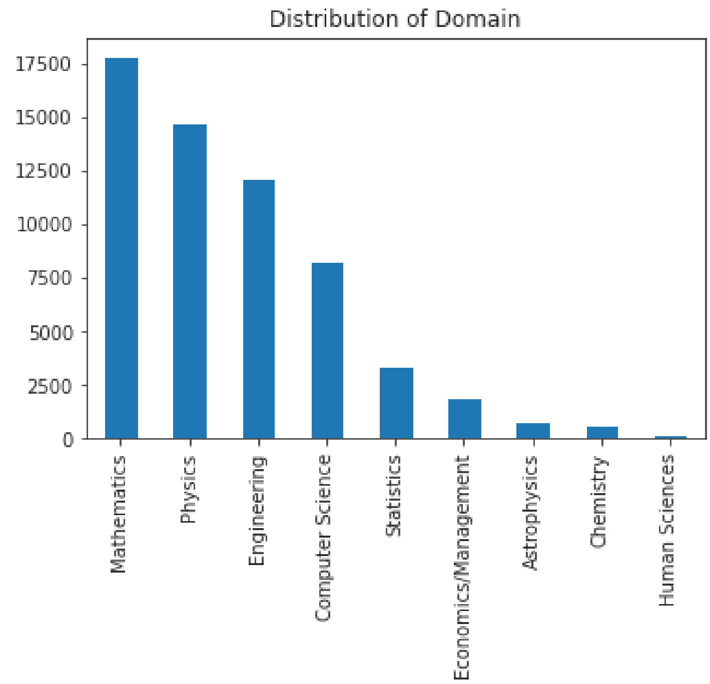
```
fn = test_text1.iloc[index]
```

In [36]:

```
plt.title('Distribution of Domain')
test_text1['Domain'].value_counts().plot(kind='bar')
```

Out[36]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f44292fc810>
```

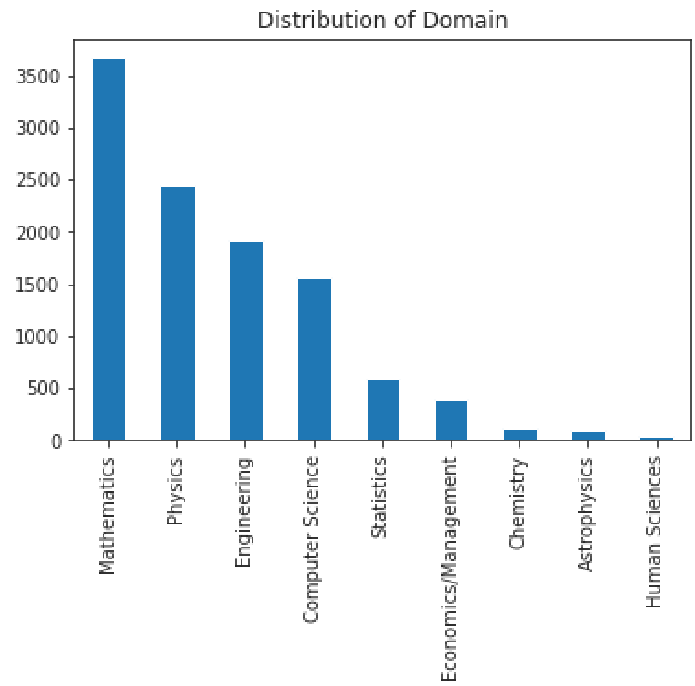


In [37]:

```
plt.title('Distribution of Domain')
fn['Domain'].value_counts().plot(kind='bar')
```

Out[37]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f44286b2d10>
```



In [38]:

```
delw = fn['del_word'].value_counts().sort_values(ascending = False).head(30)
```

In [39]:

```
delw
```

Out[39]:

```
Series([], Name: del_word, dtype: int64)
```

In [40]:

```
insw = fn['ins_word'].value_counts().sort_values(ascending = False).head(30)
```

In [41]:

```
insw
```

Out[41]:

```
Series([], Name: ins_word, dtype: int64)
```

Analysing True Negatives

In [42]:

```
index = []
for i,(l,p) in enumerate(zip(y_ts,preds)):
  if l == 0 and p ==0:
    index.append(i)
```

In [43]:

```
cm.iloc[0,0]==len(index)
```

Out[43]:

```
True
```

In [44]:

```
tn = test_text1.iloc[index]
```

In [45]:

```
tnlabels = np.take(test_labelsf[:,0],index)
```

In [46]:

```
np.unique(tnlabels)
```

Out[46]:

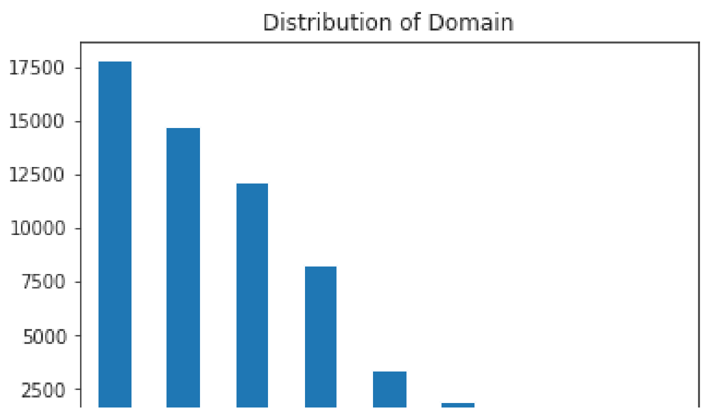
```
array([0.], dtype=float32)
```

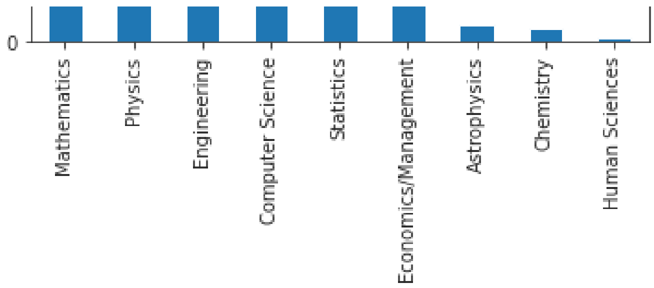
In [47]:

```
plt.title('Distribution of Domain')
test_text1['Domain'].value_counts().plot(kind='bar')
```

Out[47]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f4428eaca10>
```



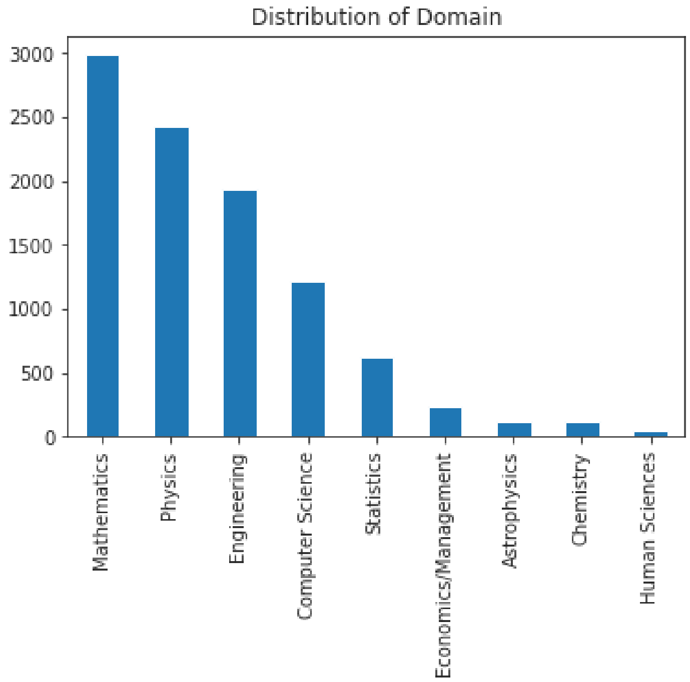


In [48]:

```
plt.title('Distribution of Domain')
tn['Domain'].value_counts().plot(kind='bar')
```

Out[48]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f4428e91310>
```



In [49]:

```
del_ins_pair = tn['del_word']+' '+tn['ins_word']
```

In [50]:

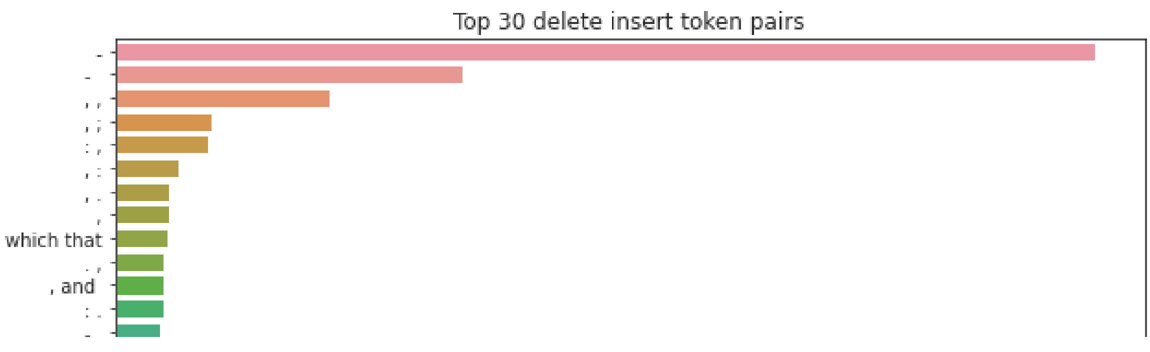
```
del_ins_pair_plt = del_ins_pair.value_counts().sort_values(ascending = False).head(30)
```

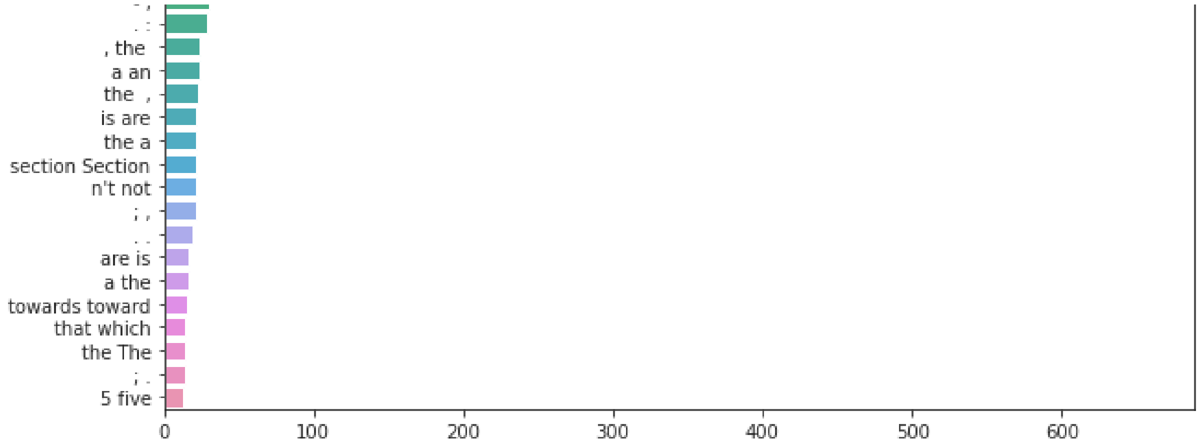
In [51]:

```
plt.figure(figsize=(10,7))
sns.set_style('ticks')
plt.title('Top 30 delete insert token pairs')
sns.barplot(y=del_ins_pair_plt.index, x= del_ins_pair_plt.values )
```

Out[51]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f44291156d0>
```





Error analysis

In [55]:

```
from tensorflow.keras.losses import binary_crossentropy

loss = []
for (l,p) in zip(y_ts,y_pr_ts):
  value = binary_crossentropy(tf.constant([l]), tf.constant([p]))
  loss.append(value)

loss = np.array(loss)
```

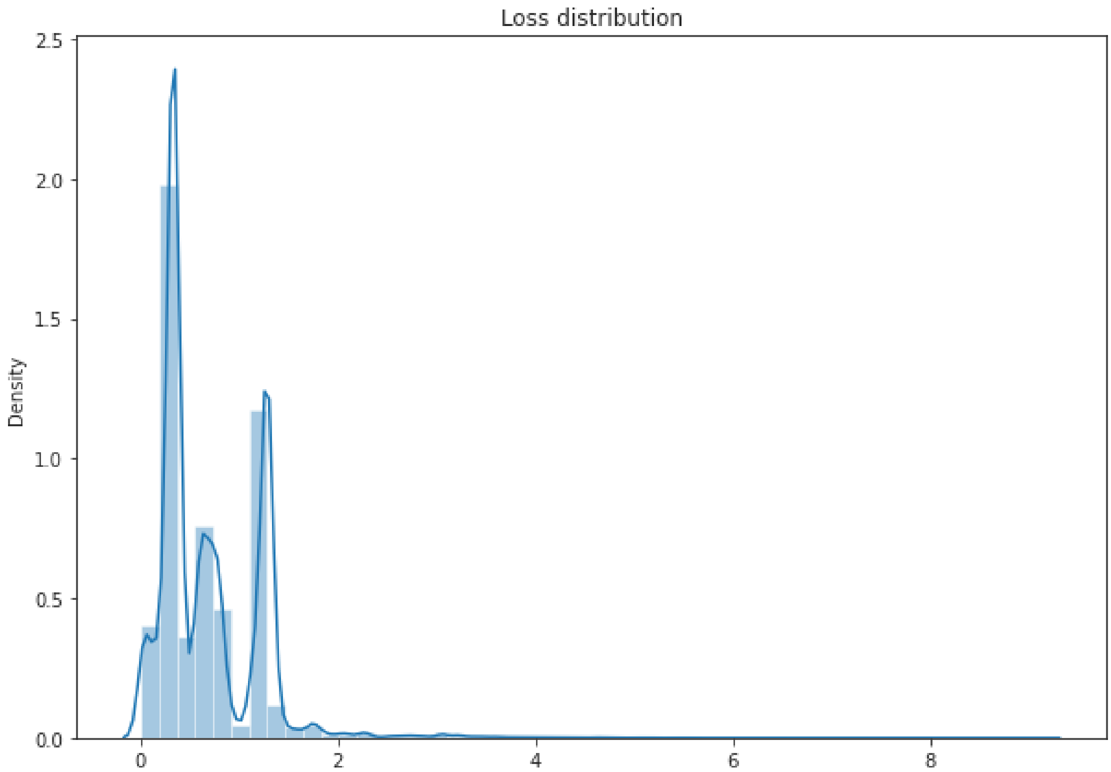
In [56]:

```
plt.figure(figsize=(10,7))
sns.set_style('ticks')
plt.title('Loss distribution')
sns.distplot(loss)
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/distributions.py:2619: FutureWarning: `dis
tplot` is a deprecated function and will be removed in a future version. Please adapt you
r code to use either `displot` (a figure-level function with similar flexibility) or `his
tplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

Out[56]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f4428ab9fd0>
```



In [57]:

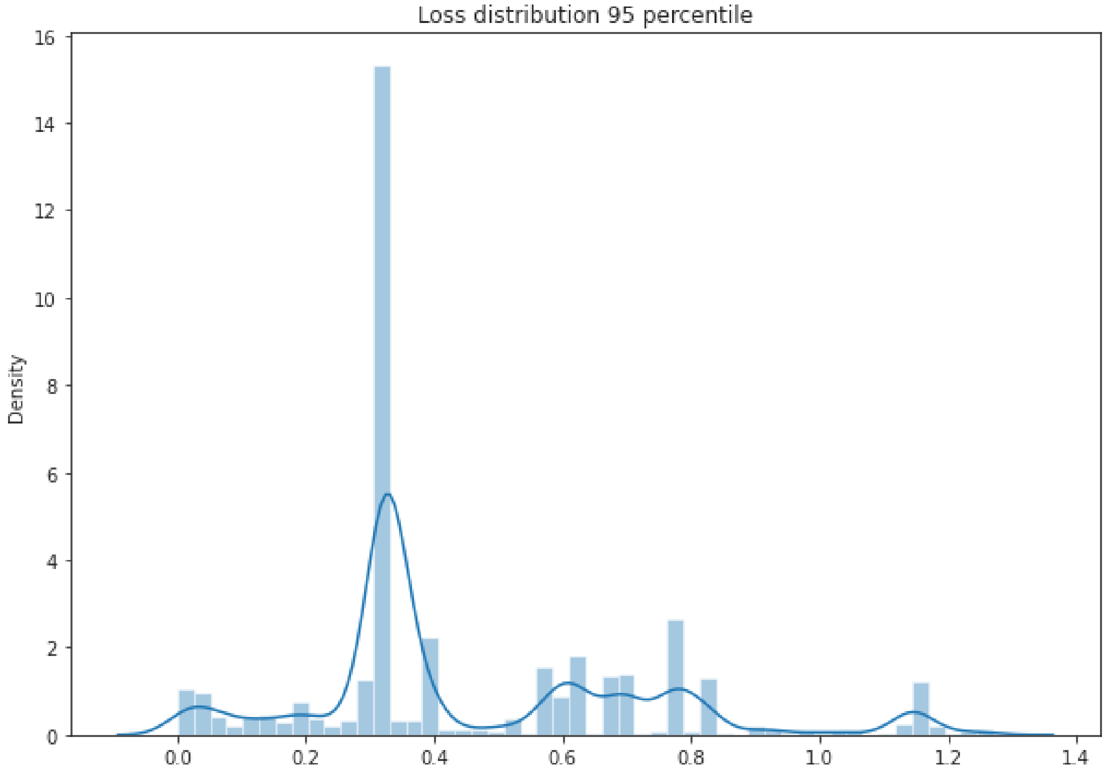
```
thres = np.percentile(loss,90)
plotl = [l for l in loss if l<thres]

plt.figure(figsize=(10,7))
sns.set_style('ticks')
plt.title('Loss distribution 95 percentile')
sns.distplot(plotl)
```

```
/usr/local/lib/python3.7/dist-packages/seaborn/distributions.py:2619: FutureWarning: `dis
tplot` is a deprecated function and will be removed in a future version. Please adapt you
r code to use either `displot` (a figure-level function with similar flexibility) or `his
tplot` (an axes-level function for histograms).
  warnings.warn(msg, FutureWarning)
```

Out[57]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f4428e0fa10>
```

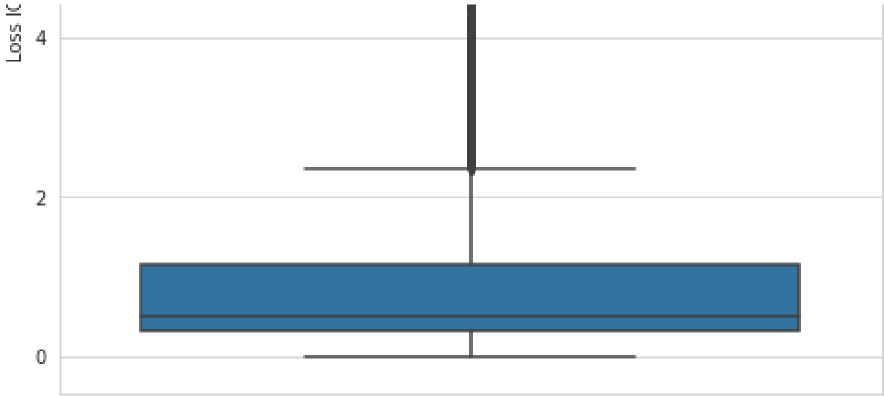


In [58]:

```
plt.figure(figsize=(8,8))
sns.set_style('whitegrid')
plt.title('Loss IQR')
sns.boxplot(y= loss).set(ylabel ='Loss IQR range')
```

Out[58]:

```
[Text(0, 0.5, 'Loss IQR range')]
```



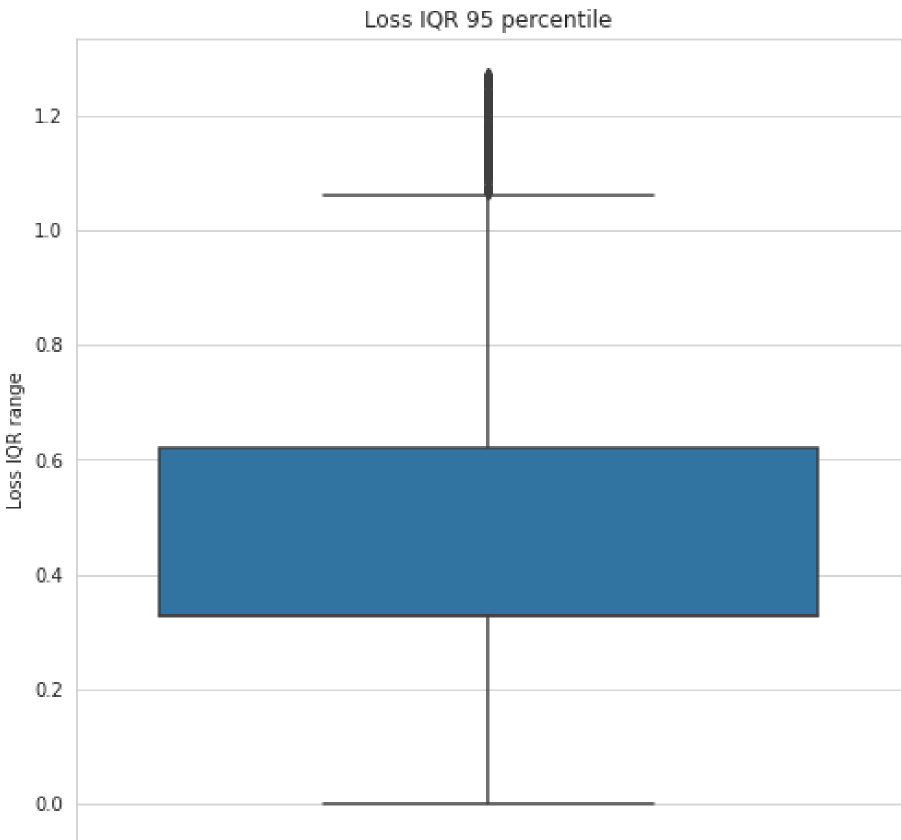
In [59]:

```
thres = np.percentile(loss,90)
plotl = [l for l in loss if l<thres]

plt.figure(figsize=(8,8))
sns.set_style('whitegrid')
plt.title('Loss IQR 95 percentile')
sns.boxplot(y= plotl).set(ylabel ='Loss IQR range')
```

Out[59]:

```
[Text(0, 0.5, 'Loss IQR range')]
```



Most Erroneous points

In [60]:

```
#getting most errornous points
thres = np.percentile(loss,90)
index = [i for i,value in enumerate(loss)  if value>=thres]
```

In [61]:

```
#Plotting confution matrix with most errornous points
erryts = np.take(y_ts,index)
errpreds = np.take(preds,index)
```

```
cm = pd.DataFrame(confusion_matrix(erryts, errpreds), range(2),range(2))
cm
```

Out[61]:

	0	1
0	0	13811
1	59	0

Most of the high erroneous points are from False Positive values i.e. the texts that need editing but are classified as otherwise.

In [62]:

```
errpoints = test_text1.iloc[index]
```

In [63]:

```
del_ins_pair = errpoints['del_word']+' '+errpoints['ins_word']
```

In [64]:

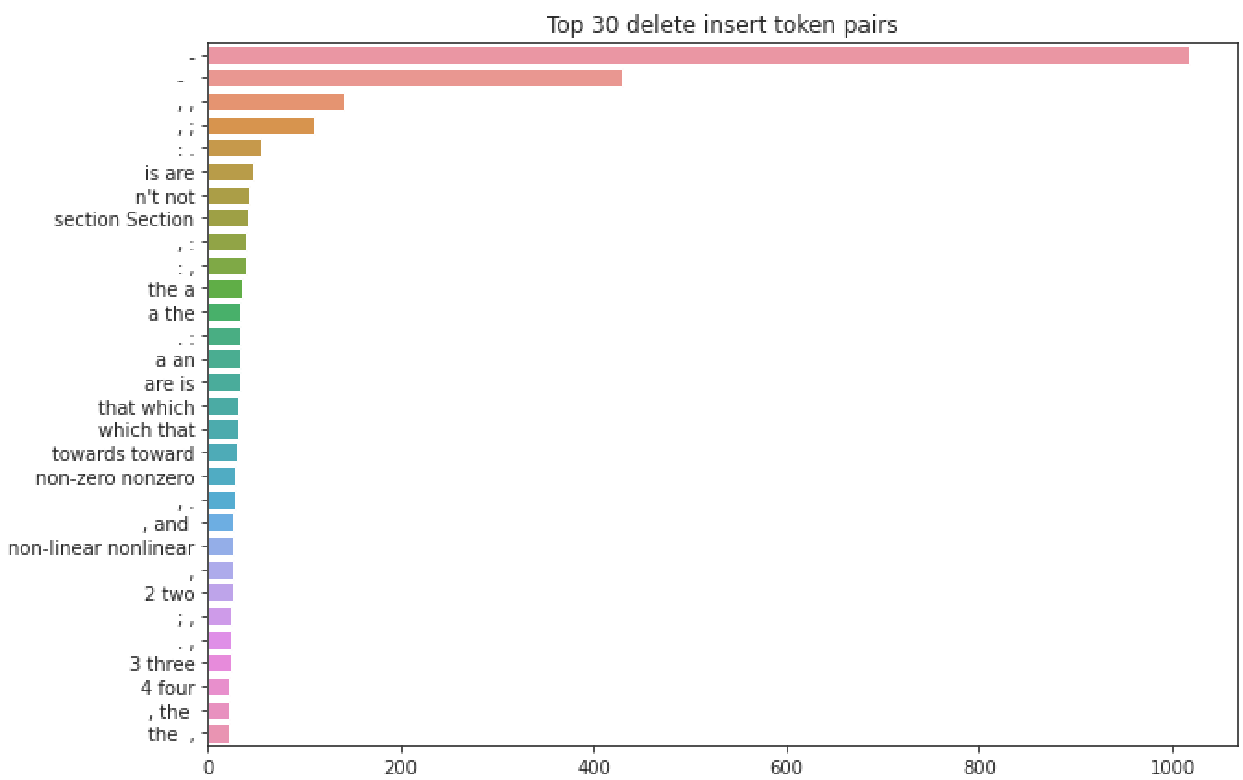
```
del_ins_pair_plt = del_ins_pair.value_counts().sort_values(ascending = False).head(30)
```

In [65]:

```
plt.figure(figsize=(10,7))
sns.set_style('ticks')
plt.title('Top 30 delete insert token pairs')
sns.barplot(y=del_ins_pair_plt.index, x= del_ins_pair_plt.values )
```

Out[65]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f44288b0ad0>
```



In [66]:

```
errpoints[['SBE','SAE','del_word','ins_word']]
```

Out[66]:

	SBE	SAE	del_word	ins_word

	SBE	SAE	del_word	ins_word
202753	The corresponding values are _MATH_ (Sept. 5, ...	The corresponding values are _MATH_ (Sept. 5, ...	Sept.	September
120881	Consequently, by taking the average of Stokes ...	Consequently, by taking the average of Stokes ...	as small as	NaN
274647	High ILP register requirements has direct impa...	High ILP register requirements have a direct i...	in	on
88330	On the other hand, due to _REF_, _MATH_ satisf...	On the other hand, due to _REF_, _MATH_ also s...	satisfies	satisfies
77630	If the nonzero _MATH_ is entirely generated by...	If the non-zero _MATH_ is entirely generated b...	nonzero	non-zero
...	...	...	...	...
64794	The comparison is shown in Table _REF_	A comparison is shown in Table _REF_.	The	.
223317	In 1998 _CITE_, it is known that our universe ...	In 1998 _CITE_, it became known that our unive...	is	became
342236	This makes it possible that more performance r...	This makes it possible for more performance re...	could	to
384417	Thus, such an accumulation generates a fund fr...	Thus, such an accumulation generates a fund fr...	NaN	,
387954	Although the above normal ordering is not uniq...	Although the above normal ordering is not uniq...	,	NaN

13870 rows × 4 columns

Least Errorenous points

In [81]:

```
#getting most errornous points
thres = np.percentile(loss,40)
index = [i for i,value in enumerate(loss)  if value<=thres]
```

In [82]:

```
#Plotting confution matrix with most errornous points
erryts = np.take(y_ts,index)
errpreds = np.take(preds,index)
cm = pd.DataFrame(confusion_matrix(erryts, errpreds), range(2),range(2))
cm
```

Out[82]:

	0	1
0	105	0
1	0	25629

In [89]:

```
errpoints = test_text1.iloc[index]
```

In [90]:

```
del_ins_pair = errpoints['del_word'].fillna(' ')+' '+errpoints['ins_word'].fillna(' ')
```

In [91]:

```
del_ins_pair_plt = del_ins_pair.value_counts().sort_values(ascending = False).head(30)
```

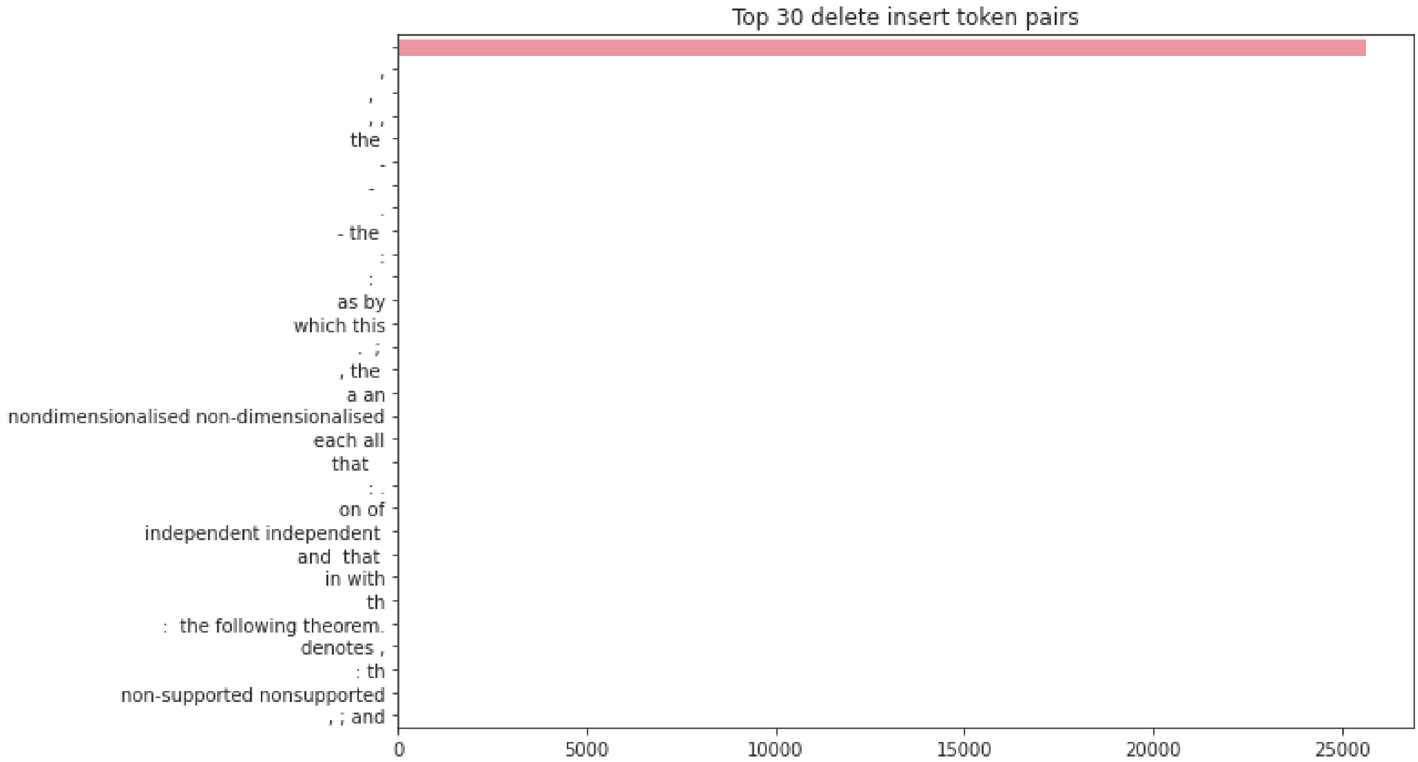
In [92]:

```
plt.figure(figsize=(10,7))
sns.set_style('ticks')
```

```
plt.title('Top 30 delete insert token pairs')
sns.barplot(y=del_ins_pair_plt.index, x= del_ins_pair_plt.values )
```

Out[92]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f4428a0db90>
```



In [93]:

```
errpoints[['SBE','SAE','del_word','ins_word']]
```

Out[93]:

	SBE	SAE	del_word	ins_word
233181	The inclusion of immune effectors reflects the...	The inclusion of immune effectors reflects the...	NaN	NaN
269482	At present, the values of the pion polarisabil...	At present, the values of the pion polarisabil...	NaN	NaN
164312	The receipt should be kept in case of disputes.	The receipt should be kept in case of disputes.	NaN	NaN
17385	Our experimental results provide valuable insi...	Our experimental results provide valuable insi...	NaN	NaN
266075	It makes sense to abstract the definition of a...	It makes sense to abstract the definition of a...	NaN	NaN
...	...	...	...	...
281543	Being based on the whole posterior distributio...	Being based on the whole posterior distributio...	NaN	NaN
258201	Note that Tables _REF_ and _REF_ were computed...	Note that Tables _REF_ and _REF_ were computed...	NaN	NaN
183644	Theorems III-IV are then deduced from Theorems...	Theorems III-IV are then deduced from Theorems...	NaN	NaN
103358	Specifically, the stimulation of ECs by VEGF l...	Specifically, the stimulation of ECs by VEGF l...	NaN	NaN
102295	The binding sites on each head and the actin f...	The binding sites on each head and the actin f...	NaN	NaN

25734 rows × 4 columns

without the NaN values

In [94]:

```
del_ins_pair = errpoints['del_word']+' '+errpoints['ins_word']
```

In [95]:

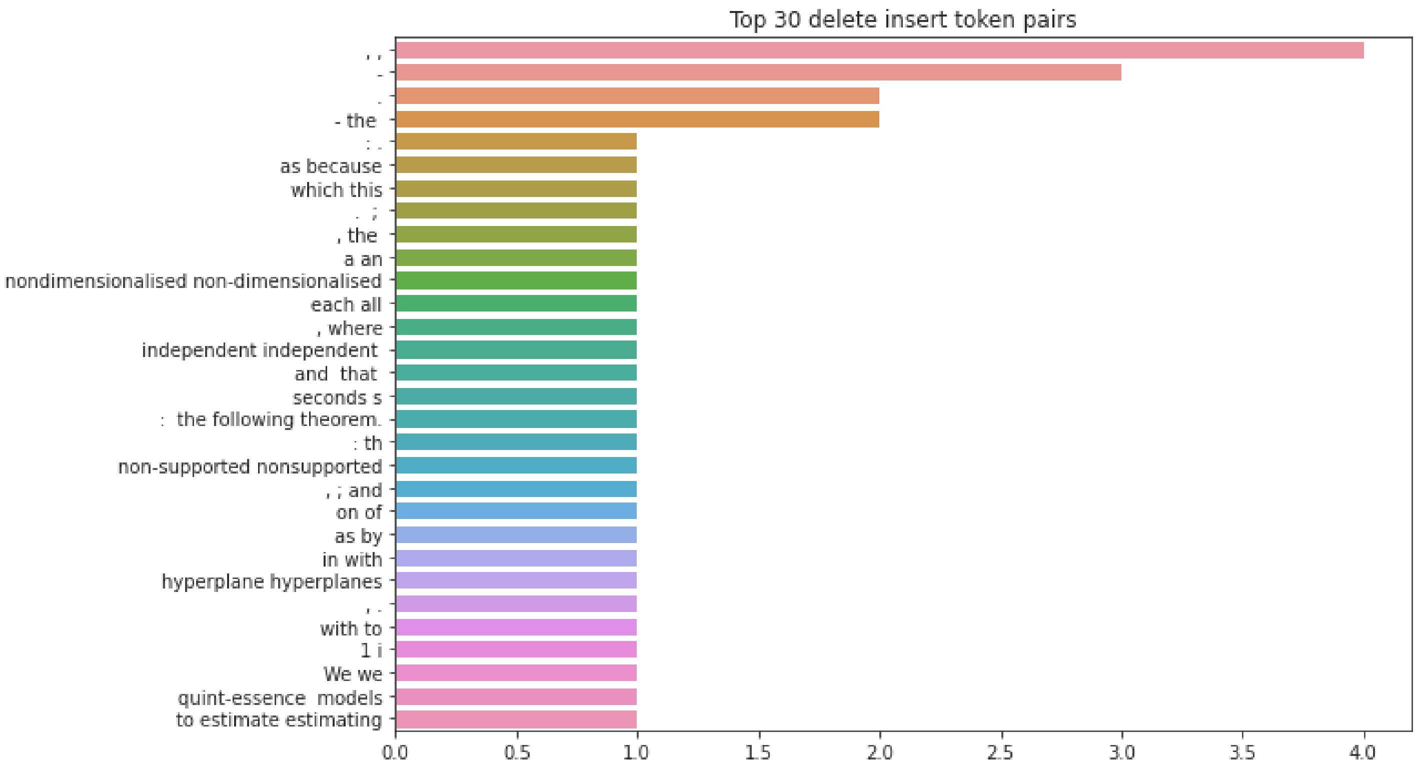
```
del_ins_pair_plt = del_ins_pair.value_counts().sort_values(ascending = False).head(30)
```

In [96]:

```
plt.figure(figsize=(10,7))
sns.set_style('ticks')
plt.title('Top 30 delete insert token pairs')
sns.barplot(y=del_ins_pair_plt.index, x= del_ins_pair_plt.values )
```

Out[96]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f44283a8f10>
```



In [97]:

```
errpoints[errpoints['del_word'].notna() & errpoints['ins_word'].notna()][['SBE','SAE','de
l_word','ins_word']]
```

Out[97]:

	SBE	SAE	del_word	ins_word
325349	If _MATH_ and _MATH_ with _MATH_ and _MATH_ in...	If _MATH_ and _MATH_ with independent _MATH_ a...	independent	independent
19647	In the meantime, a sequence _MATH_ in _MATH_ i...	In the meantime, a sequence _MATH_ in _MATH_ i...	,	,
40844	We say that a sequence of admissible control _...	We say that a sequence of admissible controls ...	,	the
399847	Since _MATH_ is unknown, we can use its estima...	Since _MATH_ is unknown, we can use its estima...	-	the
135282	This can be done as follows _MATHDISP_ with: _...	This can be done as follows: _MATHDISP_ with _...	:	th
376285	(1) If in particular _MATH_ is also a vector s...	(i) If in particular _MATH_ is also a vector s...	1	i
389796	() _MATH_; _MATH_. _MATH_; _MATH_. _MATH_; _MA...	() _MATH_, _MATH_; _MATH_, _MATH_; _MATH_, _MA...	.	;
310585	We have the following result:	We have the following result.	:	.
293417	The operator _MATH_ is said to be _MATH_-posit...	The operator _MATH_ is said to be _MATH_-posit...	on	of
375346	Let the _MATH_ transfer matrix _MATH_ be parti...	Let the _MATH_ transfer matrix _MATH_ be parti...	in	with

	SBE	SAE	del_word	ins_word
356165	However, the dependence of the dominating proc...	However, the dependence of the dominating proc...	as	because
264799	Let _MATH_ be the global mobility average in t...	Let _MATH_ be the global mobility average in t...	Where	where
385341	Fig. 6 (first column) displays the velocity ve...	Fig. 6 (first column) displays the velocity ve...	seconds	s
387259	The system matrices are given by _MATHDISP_, T...	The system matrices are given by _MATHDISP_. T...		.
101399	Let _MATHDISP_ be a sequence of constructible ...	Let _MATHDISP_ be a sequence of constructible ...		-
118698	_MATH_ data set _MATH_, _MATH_ in _MATH_, are ...	_MATH_ data set _MATH_, with _MATH_ in _MATH_,...	to estimate	estimating
86528	Using optimality (Eq. (_REF_)) to reexpress th...	Using optimality (Eq. (_REF_)) to re-express t...	reexpress	re-express
192548	For all _MATH_, the substitution of _MATH_ for...	For all _MATH_, the substitution of _MATH_ for...		.
82885	Considering the uncertain chaotic systems are ...	Considering the uncertain chaotic systems are ...	denotes	,
278797	And then, construct the adaptive control _MATH...	Then, we construct the adaptive control _MATH_...	-	the
23640	Furthermore, the relevant magnetic topology is...	Furthermore, the relevant magnetic topology is...	8	eight
344503	A code matrix _MATH_ of size _MATH_ is _MATH_-...	A code matrix _MATH_ of size _MATH_ is _MATH_-...	each	all
214991	The first factor refers to the starting matrix...	The first factor refers to the starting matrix...	,	; and
292471	There exist neighborhoods _MATH_ and _MATH_, a...	There exist neighborhoods _MATH_ and _MATH_, a...	:	, 3.
302156	The nonlinear functions _MATH_ and _MATH_ (_MA...	The nonlinear functions _MATH_ and _MATH_ (_MA...	,	and
302082	(V1) There exists a positive number _MATH_ suc...	(V1) There exists a positive number _MATH_ suc...	exist	exists
28895	The multiple measurement vectors (MMV) problem...	The multiple measurement vectors (MMV) problem...	etc.	,
237375	Then the Markov jump process _MATH_ is positiv...	Then the Markov jump process _MATH_ is positiv...		-
197327	The _MATH_ decaying into _MATH_, _MATH_, the _...	_MATH_ decaying into _MATH_, _MATH_, _MATH_, _...	with	to
37958	Minimize _MATH_ subject to _MATHDISP_ Maximize...	Minimize _MATH_ subject to _MATHDISP_; Maximiz...		.
15248	Given are matrices _MATH_, _MATH_, _MATH_, _MA...	Given are matrices _MATH_, _MATH_, _MATH_, _MA...	with	such that
124149	For some given constants _MATH_ and _MATH_, th...	For some given constants _MATH_ and _MATH_, sy...	,	,
351182	The matrices _MATH_, _MATH_, _MATH_ and _MATH_...	The matrices _MATH_, _MATH_, _MATH_ and _MATH_...	,	where
88608	For given positive constants _MATH_ and _MATH_...	For given positive constants _MATH_ and _MATH_...	,	,
45860	Then, there exists a reduced-order _MATH_ filt...	Then, there exists a reduced-order _MATH_ filt...	,	,
321819	We denote by _MATH_ the space of square-integr...	We denote by _MATH_ the space of square integr...	-	
125098	The nondimensionalised boundary conditions rea...	The non-dimensionalised boundary conditions re...	nondimensionalised	non-dimensionalised
2588	Many kinds of DE models have already been cons...	Many kinds of DE models have already been cons...	quint-essence	models
357276	Define the following data probability	Define the following data probability		when

	ratios SBE	ratios SAE	del_word	ins_word
332822	From (_REF_), (_REF_), (_REF_) and (_REF_), we...	From (_REF_), (_REF_), (_REF_), and (_REF_), w...	which	this
239383	Let _MATH_ be the (identifiable) parameter vec...	Let _MATH_ be the (identifiable) parameter vec...	,	let
139855	Under the above assumptions we have _MATHDISP_...	Under the above assumptions, we have _MATHDISP...	,	;
155327	_MATHDISP_ _MATHDISP_ _MATHDISP_ _MATHDISP_ wh...	_MATHDISP_ _MATHDISP_ _MATHDISP_ _MATHDISP_ wh...	as	by
269935	Calculating _MATH_, we get _MATHDISP_ Using Le...	Calculating _MATH_, we get _MATHDISP_ Using Le...	and	,
320022	Then the linear complexity of _MATH_ is define...	Then the linear complexity of _MATH_ is define...	non-negative	nonnegative
254494	Now, for instance, the SE_MATH_ associated wit...	Now, for instance, the SE_MATH_ associated wit...	hyperplane	hyperplanes
42685	The "if" direction trivially follows from the ...	The "if" direction trivially follows from the ...	non-supported	nonsupported
360865	In the Section 3, the partial DOE with small p...	In Section 3, the partial DOE with small param...	a	an
315183	By induction, we obtain:	By induction, we obtain the following theorem.	:	the following theorem.
343516	In particular the _MATH_ component of _MATH_, ...	In particular, the _MATH_ component of _MATH_,...	:	,
171511	For the choice _MATH_, _MATH_, _MATH_ and _MAT...	For the choice _MATH_, _MATH_, _MATH_ and _MAT...	We	we
274417	These calculations imply that in the group _MA...	These calculations imply that in the group _MA...		-
304362	Given a set of symmetric matrices _MATH_, _MAT...	Given a set of symmetric matrices _MATH_, _MAT...	and	that
185177	The reference prior, _MATH_, for _MATH_ is of ...	The reference prior, _MATH_, for _MATH_ is of ...	,	.
328346	(i) the generating function _MATH_ is such tha...	(i) The generating function _MATH_ is such tha...	the	The
88321	_MATHDISP_. _MATH_, _MATH_, _MATH_, _MATH_, _M...	_MATHDISP_, for _MATH_, _MATH_, _MATH_, _MATH_...	.	for

Conclusion

1. Distillbert Trained on 400k data points gave the best F1 score.
2. No of FP: 13811. No of Fn: 10845.
3. The data that belonged to FP had the same dist of domain as that of whole Train Class but Physics. So FP are little dependent on domain.
4. FN's domain dist followed the Test class.
5. The deleted-then-inserted word pair for FP mostly had Punctuations pairs, followed by is-are,section-Section,n't-not pairs, and then articles(a,an,the) pairs that got falsely predicted as not needing editing.
6. The deleted-then-inserted word pair for TN mostly had Punctuations pairs that got Truley predicted as needing editing.
7. Binary crossentropy loss dist is highly right skewed.
8. till 90th percentile of loss is irregularly distributed.
9. Most errorenous points belong to FP, and so deleted-then-inserted word pair for it closely follows FP.
10. Least error points belong to TN+TP and so most of the deleted-then-inserted word pair are NaN and apart from NaN values it is a subset of TN(105 points out of 10845).