Automated Classification of News using Machine Learning and Natural Language Processing.

Setting up the Notebook and Exploration of Data.

Installing Modules

> Preprocessing and Customizing

1. Numpy
2. Pandas
3. re (regular expressions)

> For Visualisation

1. Matplotlib
2. wordcloud
3. seaborn

> For Modelling

1. sklearn
2. nltk

```
import nltk
from nltk.corpus import stopwords
from nltk.stem import PorterStemmer
from nltk.tokenize import word_tokenize
import numpy as np
import pandas as pd

import matplotlib.pyplot as plt
import seaborn as sns

from tensorflow.keras.preprocessing.text import Tokenizer
from tensorflow.keras.preprocessing.sequence import pad_sequences
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score

import tensorflow as tf

from sklearn.metrics import confusion_matrix, classification_report
from wordcloud import WordCloud ,STOPWORDS
nltk.download('stopwords')
nltk.download('punkt')
nltk.download('averaged_perceptron_tagger')
from nltk.tokenize import sent_tokenize, word_tokenize
import re
```

```
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Package punkt is already up-to-date!
[nltk_data] Downloading package averaged_perceptron_tagger to
[nltk_data]     /root/nltk_data...
[nltk_data]   Package averaged_perceptron_tagger is already up-to-
[nltk_data]       date!
```

Converting the dataset into pandas dataframe

```
df = pd.read_csv('/content/drive/MyDrive/Minor-Project/dataset.csv')
df.shape
```

```
(117817, 4)
```

General view on the dataset (category - value)

```
df['label'].value_counts()
```

```
economics        9308
tech             9293
business         9258
finance          9167
beauty           9130
entertainment    9028
food             8876
sports           8876
politics         8719
science          8582
travel           8107
environment      7705
world            6211
health           5557
Name: label, dtype: int64
```

Extracting the news categories for preprocessing stage.

```
[ ]  df = df[df['label'].isin(['finance','tech','entertainment','environment','sports','politics'])]
```

```
[ ]  df['label'].unique()
```

```
array(['entertainment', 'politics', 'tech', 'finance', 'sports',
       'environment'], dtype=object)
```

```
[ ]  df.label.value_counts()
```

```
tech             9293
finance          9167
entertainment    9028
sports           8876
politics         8719
environment      7705
Name: label, dtype: int64
```

Data Preprocessing.

Data Cleaning

Steps followed in preprocessing of data:

1. Combining *title* and *excerpt* column into *comb* for training.
2. Tokenizing the merged column.
3. Removing Stopwords.
4. Clearing the Noise (Punctuation, emojies etc.)
5. Finally applied stemming with PorterStemmer function.

```
[ ]  def preprocess_inputs(df):
       df = df.copy()
       df['excerpt'] = df['excerpt'].fillna('',axis = 0)
       df['comb'] = df['title'] +' ' +df['excerpt']

       stop_words = set(stopwords.words('english'))
       df['comb_token'] = df['comb'].apply(lambda x: word_tokenize(x))
       extra = ['new','october']
       for ex in extra:
         stop_words.add(ex)
       f = lambda x: [item.lower() for item in x if item.lower() not in stop_words]
       df['comb_token'] = df['comb_token'].apply(f)

       def check_punc(word):
         if re.match(r'([A-Za-z]+)',word):
           return True
         else:
           return False

       df['comb_token'] = df['comb_token'].apply(lambda x:list(filter(check_punc,x)))

       ps = PorterStemmer()
       f = lambda arr: [ps.stem(word) for word in arr]
       df['stem'] = df['comb_token'].apply(f)

       return df
```

```
[ ]  X = preprocess_inputs(df)
```

Data Visualisation

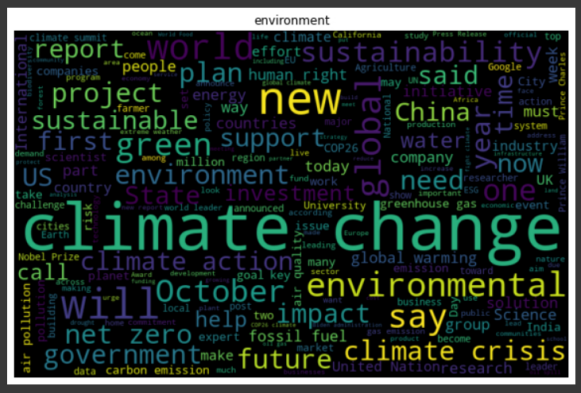
1. Checking the most frequent/relevant words for each news categories.
2. Creating wordclouds for better visualisations.

```
[ ]  def check_class(label):
       d ={}
       for x in X[X['label'] == label]['stem']:
         for y in x:
           if y.lower() in d:
             d[y.lower()]+=1
           else:
             d[y.lower()] =1

       d = {k: v for k, v in sorted(d.items(), key=lambda item: item[1], reverse=True)}
       print(label,' ---------> ',list(d.keys())[:12])

     for col in X['label'].unique():
       check_class(col)
```

```
entertainment  ---------->  ['show', 'star', 'khan', 'film', 'time', 'season', 'year', 'first', 'one', 'actor', 'say', 'get']
politics  ---------->  ['presid', 'minist', 'biden', 'trump', 'govern', 'say', 'state', 'said', 'elect', 'senat', 'report', 'vaccin']
tech  ---------->  ['announc', 'googl', 'compani', 'appl', 'technolog', 'facebook', 'launch', 'game', 'use', 'market', 'releas', 'today']
finance  ---------->  ['stock', 'inc.', 'compani', 'announc', 'share', 'bank', 'market', 'nyse', 'nasdaq', 'quarter', 'price', 'report']
sports  ---------->  ['cup', 'world', 'team', 'win', 'game', 'first', 'footbal', 'qualifi', 'sport', 'race', 'player', 'leagu']
environment  ---------->  ['climat', 'chang', 'world', 'sustain', 'global', 'environment', 'green', 'emiss', 'energi', 'action', 'carbon', 'environ']
```

```
[ ]  def create_wordcloud_visual(words,col,idx):
         wordcloud = WordCloud(width=800, height=500, random_state=21, max_font_size=110).generate(words)
         plt.figure(figsize=(10, 7))
         plt.imshow(wordcloud, interpolation="bilinear")
         plt.axis('off')
         plt.title(col)
         plt.show()
         print('\n\n')

     for idx,col in enumerate(list(X.label.unique())):
```

```
    subset=X[X.label==col]
    text= subset.title.values + ' ' + subset.excerpt.values
    words =" ".join(text)
    create_wordcloud_visual(words,col,idx)
```



Data Transformation

1. Preparing the data for modelling.
2. Spliting data into training and testing inputs.
3. Converting data into numerical form.

```
[ ]  X['final']  = X['stem'].apply(lambda arr: " ".join(arr))
```

```
[ ]  X_train, X_test, y_train, y_test = train_test_split(X['final'],
                                                         X['label'],
                                                         test_size=0.2,
                                                         random_state=42)
```

```
[ ]  from sklearn.feature_extraction.text import TfidfVectorizer
     tfidf = TfidfVectorizer(encoding='utf-8',
                             ngram_range=(1,2),
                             stop_words=None,
                             lowercase=False,
                             max_features=9000,
                             norm='l2',
                             sublinear_tf=True)


     features_train = tfidf.fit_transform(X_train).toarray()
     labels_train = y_train

     features_test = tfidf.transform(X_test).toarray()

     labels_test = y_test
```

Data Modelling.

Using Ensemble models

Random Forest Classifier

```
[ ]  from sklearn.ensemble import RandomForestClassifier
     model_rand  = RandomForestClassifier(n_estimators=100,n_jobs=-1)
```

```
model_rand.fit(features_train, labels_train)
model_predictions = model_rand.predict(features_test)
print('Accuracy: ', accuracy_score(labels_test, model_predictions))
print(classification_report(labels_test, model_predictions))
```

```
Accuracy:  0.7811138473195681
               precision    recall  f1-score   support

entertainment       0.74      0.76      0.75      1768
  environment       0.88      0.85      0.86      1551
      finance       0.76      0.74      0.75      1839
     politics       0.74      0.79      0.76      1777
       sports       0.86      0.87      0.86      1770
         tech       0.74      0.69      0.71      1853

     accuracy                           0.78     10558
    macro avg       0.78      0.78      0.78     10558
 weighted avg       0.78      0.78      0.78     10558
```

Extratrees Classifier

```
[ ]  from sklearn.ensemble import ExtraTreesClassifier
     exclf  = ExtraTreesClassifier(n_estimators=80)
     exclf.fit(features_train, labels_train)
     model_predictions = exclf.predict(features_test)
     print('Accuracy: ', accuracy_score(labels_test, model_predictions))
     print(classification_report(labels_test, model_predictions))
```

```
Accuracy:  0.7963629475279409
               precision    recall  f1-score   support

entertainment       0.76      0.76      0.76      1768
  environment       0.87      0.87      0.87      1551
      finance       0.78      0.75      0.76      1839
     politics       0.75      0.81      0.78      1777
       sports       0.87      0.89      0.88      1770
         tech       0.76      0.72      0.74      1853

     accuracy                           0.80     10558
    macro avg       0.80      0.80      0.80     10558
 weighted avg       0.80      0.80      0.80     10558
```

One vs Rest - Using SVM Model as Binary classifier

```
[ ]  from sklearn.linear_model import SGDClassifier
     sgd  = SGDClassifier()
     sgd.fit(features_train, labels_train)
     model_predictions = sgd.predict(features_test)
     print('Accuracy: ', accuracy_score(labels_test, model_predictions))
     print(classification_report(labels_test, model_predictions))
```

```
Accuracy:  0.8126539117257057
               precision    recall  f1-score   support

entertainment       0.80      0.78      0.79      1768
  environment       0.86      0.90      0.88      1551
      finance       0.82      0.72      0.77      1839
     politics       0.79      0.81      0.80      1777
       sports       0.85      0.94      0.89      1770
         tech       0.75      0.75      0.75      1853

     accuracy                           0.81     10558
    macro avg       0.81      0.82      0.81     10558
 weighted avg       0.81      0.81      0.81     10558
```

Naive_bayes - Multinomial Model

```
[ ]  from sklearn.naive_bayes import MultinomialNB
     mnb = MultinomialNB()
     mnb.fit(features_train, labels_train)
     mnb_predictions = mnb.predict(features_test)
     print('Accuracy: ', accuracy_score(labels_test, mnb_predictions))
     print(classification_report(labels_test, mnb_predictions))
```

```
Accuracy:  0.8004356885773821
               precision    recall  f1-score   support

entertainment       0.80      0.78      0.79      1768
  environment       0.82      0.87      0.85      1551
      finance       0.81      0.72      0.76      1839
     politics       0.76      0.82      0.79      1777
       sports       0.91      0.90      0.90      1770
         tech       0.72      0.73      0.73      1853

     accuracy                           0.80     10558
    macro avg       0.80      0.80      0.80     10558
 weighted avg       0.80      0.80      0.80     10558
```

Neural Network - ANN

```
[ ]  from sklearn.preprocessing import LabelEncoder
     lb = LabelEncoder()
     labels_train = lb.fit_transform(labels_train)
     labels_test =lb.transform(labels_test)


     inputs = tf.keras.Input(shape=(features_train.shape[1],))
     dense_1 = tf.keras.layers.Dense(128, activation='relu')(inputs)
     dense_2 = tf.keras.layers.Dense(128, activation='relu')(dense_1)
```

```
    outputs = tf.keras.layers.Dense(6, activation='softmax')(dense_2)


    model = tf.keras.Model(inputs, outputs)

    print(model.summary())
    tf.keras.utils.plot_model(model)

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

    history = model.fit(
        features_train,
        labels_train,
        validation_split=0.2,
        batch_size=32,
        epochs=20,
        callbacks=[
            tf.keras.callbacks.EarlyStopping(
                monitor='val_loss',
                patience=3,
            )
        ]
    )
```

```
Model: "model_2"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 input_3 (InputLayer)        [(None, 9000)]            0

 dense_6 (Dense)             (None, 128)               1152128

 dense_7 (Dense)             (None, 128)               16512

 dense_8 (Dense)             (None, 6)                 774

=================================================================
Total params: 1,169,414
Trainable params: 1,169,414
Non-trainable params: 0
_________________________________________________________________
None
Epoch 1/20
1056/1056 [==============================] - 13s 12ms/step - loss: 0.7170 - accuracy: 0.7697 - val_loss: 0.5788 - val_accuracy: 0.8121
Epoch 2/20
1056/1056 [==============================] - 12s 12ms/step - loss: 0.3850 - accuracy: 0.8725 - val_loss: 0.6230 - val_accuracy: 0.8004
Epoch 3/20
1056/1056 [==============================] - 12s 12ms/step - loss: 0.2373 - accuracy: 0.9190 - val_loss: 0.7343 - val_accuracy: 0.7891
Epoch 4/20
1056/1056 [==============================] - 12s 12ms/step - loss: 0.1232 - accuracy: 0.9583 - val_loss: 0.9546 - val_accuracy: 0.7825
```

```
[ ]  ann_predictions = model.predict([features_test])
     ann_predictions = np.argmax(ann_predictions,1)
     ann_predictions = lb.inverse_transform(ann_predictions)
     labels_test = lb.inverse_transform(labels_test)
     labels_train = lb.inverse_transform(labels_train)
     print('Accuracy: ', accuracy_score(labels_test, ann_predictions))
     print(classification_report(labels_test, ann_predictions))
```

```
Accuracy:  0.7842394392877439
               precision    recall  f1-score   support

entertainment       0.75      0.73      0.74      1768
  environment       0.86      0.85      0.85      1551
      finance       0.78      0.72      0.75      1839
     politics       0.75      0.78      0.76      1777
       sports       0.87      0.88      0.88      1770
         tech       0.70      0.75      0.73      1853

     accuracy                           0.78     10558
    macro avg       0.79      0.79      0.79     10558
 weighted avg       0.79      0.78      0.78     10558
```

Data Predictions.

Creating a custom dataset for testing the Models

```
[ ]  text_arr = [
         "LIVE BAN vs PAK 1st T20 2021 Live Cricket Score, T20 Live Match Latest Updates: Bangladesh Win Toss, Opt to Bat",
         "India vs New Zealand: Virat Kohli Sacrificed His Number 3 Position For Me, Says Suryakumar Yadav",
         "Certainly won't be playing international cricket after T20 World Cup 2022, says Matthew Wade",
         "John Abraham snatches fan's phone for filming him. Watch what happens next",
         "Amyra Dastur's Workout Routine Will Surely Be Your Motivation For The Day | Watch",
         "Salman Khan, Ranveer Singh, Shraddha Kapoor will be present at the opening ceremony of IFFI 2021",
         "Explained: All about the new annual information statement of the income tax department",
         "Personal Finance: Equity investing – It's time to rebalance your investment portfolio",
         "Why traders can get by with AI-driven platforms, but long-term investors need real advisors",
         "Recent global warming 'unprecedented' in 24,000 years, study finds",
         "COP26: India to co-lead Integrated Biorefineries Mission aimed at replacing fossil fuels",
         "51% cut in stubble burning this season against corresponding period last year: Air quality panel",
         "COVID: Germany shatters another record as Merkel, state leaders to meet",
         "Explained: What happened at Joe Biden-Xi Jinping summit, and why it matters to India",
         "PM Modi to host President Putin for one-to-one meeting on Dec 6",
         "Virtual reality has potential to change airway management training",
         "Scientists Build 256-Qubit Quantum Computer That's Unlike Anything Else",
         "Google Cloud teams up with NLP startup Cohere on multiyear partnership around TPUs"
     ]
```

```
[ ]  text_df = pd.DataFrame(text_arr,columns=['news'])
```

Functions to transform the news for model inputs

```
import re
def convert(df,tfidf,model):
  df = df.copy()
  stop_words = set(stopwords.words('english'))
  df['news_token'] = df['news'].apply(lambda x: word_tokenize(x))
  f = lambda x: [item.lower() for item in x if item.lower() not in stop_words]
  df['news_token'] = df['news_token'].apply(f)

  def check_punc(word):
    if re.match(r'([A-Za-z]+)',word):
      return True
    else:
      return False

  df['news_token'] = df['news_token'].apply(lambda x:list(filter(check_punc,x)))
  # df['comb_token'] = df['title'].apply(lambda x: blankline_tokenize(x))
  ps = PorterStemmer()
  f = lambda arr: [ps.stem(word) for word in arr]
  df['news_token'] = df['news_token'].apply(f)

  df['news_token'] = df['news_token'].apply(lambda arr: " ".join(arr))
  df['tfidf'] = df['news_token'].apply(lambda x: tfidf.transform(pd.Series(x)).toarray())
  df['type'] = df['tfidf'].apply(lambda x: model.predict(x))
  df =df.drop(['news_token','tfidf'],axis =1)
  return df
```

Predicting the news.

```
pd.set_option("display.max_colwidth", -1)
```

```
/usr/local/lib/python3.7/dist-packages/ipykernel_launcher.py:1: FutureWarning: Passing a negative integer is deprecated in version 1.0 and will not be supported in future version. Inste
  """Entry point for launching an IPython kernel.
```

```
ans = convert(text_df,tfidf,model_rand)
ans
```

	news	type
0	LIVE BAN vs PAK 1st T20 2021 Live Cricket Score, T20 Live Match Latest Updates: Bangladesh Win Toss, Opt to Bat	[sports]
1	India vs New Zealand: Virat Kohli Sacrificed His Number 3 Position For Me, Says Suryakumar Yadav	[sports]
2	Certainly won't be playing international cricket after T20 World Cup 2022, says Matthew Wade	[sports]
3	John Abraham snatches fan's phone for filming him. Watch what happens next	[entertainment]
4	Amyra Dastur's Workout Routine Will Surely Be Your Motivation For The Day | Watch	[entertainment]
5	Salman Khan, Ranveer Singh, Shraddha Kapoor will be present at the opening ceremony of IFFI 2021	[entertainment]
6	Explained: All about the new annual information statement of the income tax department	[politics]
7	Personal Finance: Equity investing — It's time to rebalance your investment portfolio	[finance]
8	Why traders can get by with AI-driven platforms, but long-term investors need real advisors	[finance]
9	Recent global warming 'unprecedented' in 24,000 years, study finds	[environment]
10	COP26: India to co-lead Integrated Biorefineries Mission aimed at replacing fossil fuels	[environment]
11	51% cut in stubble burning this season against corresponding period last year: Air quality panel	[entertainment]
12	COVID: Germany shatters another record as Merkel, state leaders to meet	[politics]
13	Explained: What happened at Joe Biden-Xi Jinping summit, and why it matters to India	[politics]
14	PM Modi to host President Putin for one-to-one meeting on Dec 6	[politics]
15	Virtual reality has potential to change airway management training	[politics]
16	Scientists Build 256-Qubit Quantum Computer That's Unlike Anything Else	[tech]
17	Google Cloud teams up with NLP startup Cohere on multiyear partnership around TPUs	[tech]

```
ans = convert(text_df,tfidf,mnb)
ans
```

	news	type
0	LIVE BAN vs PAK 1st T20 2021 Live Cricket Score, T20 Live Match Latest Updates: Bangladesh Win Toss, Opt to Bat	[sports]
1	India vs New Zealand: Virat Kohli Sacrificed His Number 3 Position For Me, Says Suryakumar Yadav	[sports]
2	Certainly won't be playing international cricket after T20 World Cup 2022, says Matthew Wade	[sports]
3	John Abraham snatches fan's phone for filming him. Watch what happens next	[entertainment]
4	Amyra Dastur's Workout Routine Will Surely Be Your Motivation For The Day | Watch	[entertainment]
5	Salman Khan, Ranveer Singh, Shraddha Kapoor will be present at the opening ceremony of IFFI 2021	[entertainment]
6	Explained: All about the new annual information statement of the income tax department	[finance]
7	Personal Finance: Equity investing — It's time to rebalance your investment portfolio	[finance]
8	Why traders can get by with AI-driven platforms, but long-term investors need real advisors	[finance]
9	Recent global warming 'unprecedented' in 24,000 years, study finds	[environment]
10	COP26: India to co-lead Integrated Biorefineries Mission aimed at replacing fossil fuels	[environment]
11	51% cut in stubble burning this season against corresponding period last year: Air quality panel	[environment]
12	COVID: Germany shatters another record as Merkel, state leaders to meet	[politics]

13	Explained: What happened at Joe Biden-Xi Jinping summit, and why it matters to India	[politics]
14	PM Modi to host President Putin for one-to-one meeting on Dec 6	[politics]
15	Virtual reality has potential to change airway management training	[tech]
16	Scientists Build 256-Qubit Quantum Computer That's Unlike Anything Else	[tech]
17	Google Cloud teams up with NLP startup Cohere on multiyear partnership around TPUs	[tech]

```
[ ]  ans = convert(text_df,tfidf,model)
     ans['type'] = ans['type'].apply(lambda arr: lb.inverse_transform(np.argmax(arr,1)))
     ans
```

	news	type
0	LIVE BAN vs PAK 1st T20 2021 Live Cricket Score, T20 Live Match Latest Updates: Bangladesh Win Toss, Opt to Bat	[sports]
1	India vs New Zealand: Virat Kohli Sacrificed His Number 3 Position For Me, Says Suryakumar Yadav	[sports]
2	Certainly won't be playing international cricket after T20 World Cup 2022, says Matthew Wade	[sports]
3	John Abraham snatches fan's phone for filming him. Watch what happens next	[entertainment]
4	Amyra Dastur's Workout Routine Will Surely Be Your Motivation For The Day | Watch	[entertainment]
5	Salman Khan, Ranveer Singh, Shraddha Kapoor will be present at the opening ceremony of IFFI 2021	[entertainment]
6	Explained: All about the new annual information statement of the income tax department	[politics]
7	Personal Finance: Equity investing — It's time to rebalance your investment portfolio	[finance]
8	Why traders can get by with AI-driven platforms, but long-term investors need real advisors	[finance]
9	Recent global warming 'unprecedented' in 24,000 years, study finds	[environment]
10	COP26: India to co-lead Integrated Biorefineries Mission aimed at replacing fossil fuels	[environment]
11	51% cut in stubble burning this season against corresponding period last year: Air quality panel	[environment]
12	COVID: Germany shatters another record as Merkel, state leaders to meet	[politics]
13	Explained: What happened at Joe Biden-Xi Jinping summit, and why it matters to India	[politics]
14	PM Modi to host President Putin for one-to-one meeting on Dec 6	[entertainment]
15	Virtual reality has potential to change airway management training	[sports]
16	Scientists Build 256-Qubit Quantum Computer That's Unlike Anything Else	[tech]
17	Google Cloud teams up with NLP startup Cohere on multiyear partnership around TPUs	[tech]

```
[ ]  ans = convert(text_df,tfidf,exclf)
     ans
```

	news	type
0	LIVE BAN vs PAK 1st T20 2021 Live Cricket Score, T20 Live Match Latest Updates: Bangladesh Win Toss, Opt to Bat	[sports]
1	India vs New Zealand: Virat Kohli Sacrificed His Number 3 Position For Me, Says Suryakumar Yadav	[sports]
2	Certainly won't be playing international cricket after T20 World Cup 2022, says Matthew Wade	[sports]
3	John Abraham snatches fan's phone for filming him. Watch what happens next	[entertainment]
4	Amyra Dastur's Workout Routine Will Surely Be Your Motivation For The Day | Watch	[entertainment]
5	Salman Khan, Ranveer Singh, Shraddha Kapoor will be present at the opening ceremony of IFFI 2021	[entertainment]
6	Explained: All about the new annual information statement of the income tax department	[politics]
7	Personal Finance: Equity investing — It's time to rebalance your investment portfolio	[finance]
8	Why traders can get by with AI-driven platforms, but long-term investors need real advisors	[finance]
9	Recent global warming 'unprecedented' in 24,000 years, study finds	[environment]
10	COP26: India to co-lead Integrated Biorefineries Mission aimed at replacing fossil fuels	[environment]
11	51% cut in stubble burning this season against corresponding period last year: Air quality panel	[environment]
12	COVID: Germany shatters another record as Merkel, state leaders to meet	[politics]
13	Explained: What happened at Joe Biden-Xi Jinping summit, and why it matters to India	[politics]
14	PM Modi to host President Putin for one-to-one meeting on Dec 6	[politics]
15	Virtual reality has potential to change airway management training	[politics]
16	Scientists Build 256-Qubit Quantum Computer That's Unlike Anything Else	[tech]
17	Google Cloud teams up with NLP startup Cohere on multiyear partnership around TPUs	[tech]

```
[ ]  ans = convert(text_df,tfidf,sgd)
     ans
```

	news	type
0	LIVE BAN vs PAK 1st T20 2021 Live Cricket Score, T20 Live Match Latest Updates: Bangladesh Win Toss, Opt to Bat	[sports]
1	India vs New Zealand: Virat Kohli Sacrificed His Number 3 Position For Me, Says Suryakumar Yadav	[sports]
2	Certainly won't be playing international cricket after T20 World Cup 2022, says Matthew Wade	[sports]
3	John Abraham snatches fan's phone for filming him. Watch what happens next	[entertainment]
4	Amyra Dastur's Workout Routine Will Surely Be Your Motivation For The Day | Watch	[entertainment]
5	Salman Khan, Ranveer Singh, Shraddha Kapoor will be present at the opening ceremony of IFFI 2021	[entertainment]
6	Explained: All about the new annual information statement of the income tax department	[finance]
7	Personal Finance: Equity investing — It's time to rebalance your investment portfolio	[finance]
8	Why traders can get by with AI-driven platforms, but long-term investors need real advisors	[finance]
9	Recent global warming 'unprecedented' in 24,000 years, study finds	[environment]
10	COP26: India to co-lead Integrated Biorefineries Mission aimed at replacing fossil fuels	[environment]

11	51% cut in stubble burning this season against corresponding period last year: Air quality panel	[environment]
12	COVID: Germany shatters another record as Merkel, state leaders to meet	[politics]
13	Explained: What happened at Joe Biden-Xi Jinping summit, and why it matters to India	[politics]
14	PM Modi to host President Putin for one-to-one meeting on Dec 6	[politics]
15	Virtual reality has potential to change airway management training	[sports]
16	Scientists Build 256-Qubit Quantum Computer That's Unlike Anything Else	[tech]
17	Google Cloud teams up with NLP startup Cohere on multiyear partnership around TPUs	[tech]

```
[ ]
```

✓ 0s completed at 4:07 PM