Course Project

Packages and Imports

In [144]:
```
import pandas as pd
import numpy as np
import seaborn as sns
import json

import matplotlib.pyplot as plt
from sklearn.feature_extraction.text import CountVectorizer, TfidfVector
izer
from sklearn import feature_extraction, linear_model, model_selection, p
reprocessing
from sklearn.multiclass import OneVsRestClassifier
from sklearn.metrics import accuracy_score
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import TfidfTransformer
from sklearn.pipeline import Pipeline
from sklearn.svm import LinearSVC
from sklearn.linear_model import LogisticRegression
from sklearn.naive_bayes import GaussianNB,BernoulliNB,MultinomialNB,Mul
tinomialNB

from sklearn.ensemble import GradientBoostingClassifier
from sklearn.tree import DecisionTreeClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.linear_model import SGDClassifier
from sklearn.ensemble import RandomForestClassifier
from sklearn.preprocessing import LabelEncoder, MultiLabelBinarizer
from sklearn.gaussian_process import GaussianProcessClassifier
from keras.models import Model
from keras.layers import LSTM, Activation, Dense, Dropout, Input, Embedd
ing
from keras.optimizers import RMSprop
from keras.preprocessing.text import Tokenizer
from keras.preprocessing import sequence
from keras.callbacks import EarlyStopping
from keras.utils import to_categorical

from sklearn.metrics import accuracy_score, confusion_matrix,classificat
ion_report, f1_score

from tensorflow.keras.utils import plot_model

import nltk as nlp
from nltk.corpus import stopwords
import string
import re
```

Data Import

In [91]:
```
movie_data1 = pd.read_csv("movies_metadata.csv")
movie_data1['id'] = movie_data1['id'].astype(int)
```

```
/Users/jig728/opt/anaconda3/lib/python3.7/site-packages/IPython/core/in
teractiveshell.py:3063: DtypeWarning: Columns (21) have mixed types.Spe
cify dtype option on import or set low_memory=False.
  interactivity=interactivity, compiler=compiler, result=result)
```

In [92]:
```
keyword_data1 = pd.read_csv("movies_keyword.csv")
keyword_data1['id'] = keyword_data1['id'].astype(int)
```

In [93]:
```
data1 = movie_data1.join(keyword_data1.set_index('id'), on='id')
data1 = data1.drop(columns=['adult', 'belongs_to_collection',
                            'budget', 'homepage', 'imdb_id', 'original_lan
guage',
                            'popularity', 'poster_path', 'production_compa
nies',
                            'production_countries', 'release_date', 'reven
ue',
                            'runtime', 'spoken_languages', 'status', 'tagl
ine',
                            'video', 'vote_average', 'vote_count', 'origin
al_title', 'id'])
data1.drop_duplicates(inplace=True)
data1.sort_values('title').head(10)
```

Out[93]:

	genres	overview	title	keywords
23841	[{'id': 99, 'name': 'Documentary'}]	Through intimate interviews, provocative art, ...	!Women Art Revolution	[{'id': 2383, 'name': 'feminism'}, {'id': 1870...
28619	[{'id': 35, 'name': 'Comedy'}, {'id': 18, 'nam...	A pair of horny college guys get summer jobs a...	#1 Cheerleader Camp	[{'id': 6075, 'name': 'sport'}]
4385	[{'id': 18, 'name': 'Drama'}, {'id': 9648, 'na...	Inspired by actual events, a group of 12 year ...	#Horror	[]
6030	[{'id': 99, 'name': 'Documentary'}]	From her childhood bedroom in the Chicago subu...	#chicagoGirl	[]
12146	[{'id': 37, 'name': 'Western'}]	Johnny Liston has just been released from pris...	$1,000 on the Black	[]
11302	[{'id': 18, 'name': 'Drama'}, {'id': 37, 'name...	A stranger rides into Rainbow Valley where he'...	$100,000 for Ringo	[{'id': 156212, 'name': 'spaghetti western'}, ...
43536	[{'id': 18, 'name': 'Drama'}, {'id': 35, 'name...	After being released from jail, the son of a c...	$5 a Day	[]
7155	[{'id': 18, 'name': 'Drama'}]	When Ross is diagnosed with terminal brain can...	$50K and a Call Girl: A Love Story	[]
36662	[{'id': 16, 'name': 'Animation'}, {'id': 18, '...	Have you ever wondered "What is the meaning of...	$9.99	[{'id': 10183, 'name': 'independent film'}, {'...
16586	[{'id': 99, 'name': 'Documentary'}]	Fame today is more than an obsession. Fame has...	$ellebrity	[{'id': 208403, 'name': 'celebrity photographe...

In [94]:
```
movie_data2 = pd.read_csv("wiki_movie_plots_deduped.csv")
data2 =  movie_data2.drop(columns=['Release Year', 'Origin/Ethnicity',
'Director', 'Cast', 'Wiki Page'])
data2.drop_duplicates(inplace=True)
data2.sort_values('Title').head(10)
data2['Genre'] = data2['Genre'].apply(lambda x: x.split(', '))
```

In [95]:
```
movie_data3 = pd.read_csv("IMDb_movies.csv")
data3 = movie_data3.drop(columns=['imdb_title_id', 'original_title', 'ye
ar', 'date_published',
       'duration', 'country', 'language', 'director', 'writer',
       'production_company', 'actors', 'avg_vote', 'votes',
       'budget', 'usa_gross_income', 'worlwide_gross_income', 'metascor
e',
       'reviews_from_users', 'reviews_from_critics'])
data3.drop_duplicates(inplace=True)
data3.sort_values('title').head(10)
data3['genre'] = data3['genre'].apply(lambda x: x.split(', '))
```

```
/Users/jig728/opt/anaconda3/lib/python3.7/site-packages/IPython/core/in
teractiveshell.py:3063: DtypeWarning: Columns (3) have mixed types.Spec
ify dtype option on import or set low_memory=False.
  interactivity=interactivity, compiler=compiler, result=result)
```

In [96]:
```
def json_extract(obj, value):
    arr = []
    try:
        obj = eval(obj)
    except:
        return arr
    def extract(obj, arr, value):
        if isinstance(obj, dict):
            for k, v in obj.items():
                if isinstance(v, (dict, list)):
                    extract(v, arr, value)
                elif k == value:
                    arr.append(v)
        elif isinstance(obj, list):
            for item in obj:
                extract(item, arr, value)
        return arr

    values = extract(obj, arr, value)
    return values
```

In [97]:
```
data1['genres'] = data1['genres'].apply(lambda x: json_extract(x, 'name'
))
data1['keywords'] = data1['keywords'].apply(lambda x: json_extract(x, 'n
ame'))
```

In [98]:
```
data.head(10)
```

Out[98]:

	genres	overview	title	keywords
0	[Fantasy, Drama]	Manuel is a young boy who travels from long ag...	Manuel on the Island of Wonders	[]
1	[Romance, Comedy]	NaN	Thick Lashes of Lauri Mäntyvaara	[fantasy, youth, weird]
2	[Drama, Romance]	In the 1910s, beautiful young Silja loses both...	Silja - nuorena nukkunut	[]
3	[Drama]	Fifteen-year-old girl Dotty Fisher is assaulte...	Tragedy in a Temporary Town	[]
4	[Fantasy, Drama]	A horror comedy spoofing conspiracy theory mov...	Abduction	[]
5	[Documentary]	An interview session with Arnold Schwarzenegge...	The Making of 'The Terminator': A Retrospective	[making of]
6	[Documentary]	William Shatner sits down with scientists, inn...	The Truth Is in the Stars	[nature, science, canadian movie]
7	[Horror, Science Fiction]	Stranded in an Arctic mine, two survivors are ...	Zygote	[]
8	[Action, Adventure, Crime]	International master thief, Simon Templar, als...	The Saint	[the saint]
9	[Action, Science Fiction, War]	Set during the Vietnam war, Firebase follows A...	Firebase	[vietnam war, short]

In [99]:
```
data2.head(10)
```

Out[99]:

	Title	Genre	Plot
0	Kansas Saloon Smashers	[unknown]	A bartender is working at a saloon, serving dr...
1	Love by the Light of the Moon	[unknown]	The moon, painted with a smiling face hangs ov...
2	The Martyred Presidents	[unknown]	The film, just over a minute long, is composed...
3	Terrible Teddy, the Grizzly King	[unknown]	Lasting just 61 seconds and consisting of two ...
4	Jack and the Beanstalk	[unknown]	The earliest known adaptation of the classic f...
5	Alice in Wonderland	[unknown]	Alice follows a large white rabbit down a "Rab...
6	The Great Train Robbery	[western]	The film opens with two bandits breaking into ...
7	The Suburbanite	[comedy]	The film is about a family who move to the sub...
8	The Little Train Robbery	[unknown]	The opening scene shows the interior of the ro...
9	The Night Before Christmas	[unknown]	Scenes are introduced using lines of the poem....

In [100]:
```
data3.head(10)
```

Out[100]:

	title	genre	description
0	Miss Jerry	[Romance]	The adventures of a female reporter in the 1890s.
1	The Story of the Kelly Gang	[Biography, Crime, Drama]	True story of notorious Australian outlaw Ned ...
2	Den sorte drøm	[Drama]	Two men of high rank are both wooing the beaut...
3	Cleopatra	[Drama, History]	The fabled queen of Egypt's affair with Roman ...
4	L'Inferno	[Adventure, Drama, Fantasy]	Loosely adapted from Dante's Divine Comedy and...
5	From the Manger to the Cross; or, Jesus of Naz...	[Biography, Drama]	An account of the life of Jesus Christ, based ...
6	Madame DuBarry	[Biography, Drama, Romance]	The story of Madame DuBarry, the mistress of L...
7	Quo Vadis?	[Drama, History]	An epic Italian film "Quo Vadis" influenced ma...
8	Independenta Romaniei	[History, War]	The movie depicts the Romanian War of Independ...
9	Richard III	[Drama]	Richard of Gloucester uses manipulation and mu...

In [101]:
```
a = data1[data1['overview'].astype(str).map(len) >=10]
b = a[a['genres'].str.len() > 0]
b
```

Out[101]:

	genres	overview	title	keywords
0	[Fantasy, Drama]	Manuel is a young boy who travels from long ag...	Manuel on the Island of Wonders	[]
2	[Drama, Romance]	In the 1910s, beautiful young Silja loses both...	Silja - nuorena nukkunut	[]
3	[Drama]	Fifteen-year-old girl Dotty Fisher is assaulte...	Tragedy in a Temporary Town	[]
4	[Fantasy, Drama]	A horror comedy spoofing conspiracy theory mov...	Abduction	[]
5	[Documentary]	An interview session with Arnold Schwarzenegge...	The Making of 'The Terminator': A Retrospective	[making of]
...	...	...	...	...
45458	[Adventure, Action, Science Fiction]	Princess Leia is captured and held hostage by ...	Star Wars	[android, galaxy, hermit, death star, lightsab...
45459	[Action, Thriller, Crime]	While racing to a boxing match, Frank, Mike, J...	Judgment Night	[chicago, drug dealer, boxing match, escape, o...
45460	[Crime, Comedy]	It's Ted the Bellhop's first night on the job....	Four Rooms	[hotel, new year's eve, witch, bet, hotel room...
45461	[Drama, Comedy]	An episode in the life of Nikander, a garbage ...	Shadows in Paradise	[salesclerk, helsinki, garbage, independent film]
45462	[Drama, Crime]	Taisto Kasurinen is a Finnish coal miner whose...	Ariel	[underdog, prison, factory worker, prisoner, h...

42288 rows × 4 columns

In [102]:
```
b['clean_overview'] = b['overview'].apply(lambda text: clean_text(text))
b['clean_overview'] = b['clean_overview'].apply(lambda text: remove_stop
words(text))
```

```
/Users/jig728/opt/anaconda3/lib/python3.7/site-packages/ipykernel_launc
her.py:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  """Entry point for launching an IPython kernel.
/Users/jig728/opt/anaconda3/lib/python3.7/site-packages/ipykernel_launc
her.py:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
```

In [172]:
```
multilabel_binarizer = MultiLabelBinarizer()
multilabel_binarizer.fit(b['genres'])

y = multilabel_binarizer.transform(b['genres'])
tfidf_vectorizer = TfidfVectorizer()
```

In [173]:
```
y[0]
```

Out[173]:
```
array([0, 0, 0, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0])
```

In [174]:
```
x_train1,x_test1,y_train1,y_test1 = train_test_split(b['clean_overview'
], y, test_size=0.2, random_state=2020)
# x_train2,x_test2,y_train2,y_test2 = train_test_split(data2['Plot'], da
ta2['Genre'], test_size=0.2, random_state=2020)
# x_train3,x_test3,y_train3,y_test3 = train_test_split(data3['descriptio
n'], data3['genre'], test_size=0.2, random_state=2020)

xtrain_tfidf = tfidf_vectorizer.fit_transform(x_train1)
xval_tfidf = tfidf_vectorizer.transform(x_test1)

lr = LogisticRegression()
clf = OneVsRestClassifier(lr)

model = clf.fit(xtrain_tfidf, y_train1)
prediction = model.predict(xval_tfidf > .14).astype(int)
f1_score(y_test1, prediction, average="micro")
```

Out[174]:
```
0.5106127167630058
```

In [134]:
```
svc = LinearSVC()
clf = OneVsRestClassifier(svc)

model = clf.fit(xtrain_tfidf, y_train1)
prediction = model.predict(xval_tfidf)
f1_score(y_test1, prediction, average="micro")
```

Out[134]: 0.5231140379474474

In [137]:
```
mnb = MultinomialNB()
clf = OneVsRestClassifier(mnb)

model = clf.fit(xtrain_tfidf, y_train1)
prediction = model.predict(xval_tfidf)
f1_score(y_test1, prediction, average="micro")
```

Out[137]: 0.3751861042183623

In [138]:
```
bnb = BernoulliNB()
clf = OneVsRestClassifier(bnb)

model = clf.fit(xtrain_tfidf, y_train1)
prediction = model.predict(xval_tfidf)
f1_score(y_test1, prediction, average="micro")
```

Out[138]: 0.5466639891560822

In [140]:
```
gbc = GradientBoostingClassifier(loss = 'deviance', learning_rate = .01,
n_estimators = 10, max_depth=5, random_state=2020)
clf = OneVsRestClassifier(gbc)

model = clf.fit(xtrain_tfidf, y_train1)
prediction = model.predict(xval_tfidf)
f1_score(y_test1, prediction, average="micro")
```

Out[140]: 0.02650668121079902

In [141]:
```
sgd = SGDClassifier()
clf = OneVsRestClassifier(sgd)

model = clf.fit(xtrain_tfidf, y_train1)
prediction = model.predict(xval_tfidf)
f1_score(y_test1, prediction, average="micro")
```

Out[141]: 0.4028003907521979

In [143]:
```
knn = KNeighborsClassifier(n_neighbors = 10, weights = 'distance', algor
ithm = 'brute')
clf = OneVsRestClassifier(knn)

model = clf.fit(xtrain_tfidf, y_train1)
prediction = model.predict(xval_tfidf)
f1_score(y_test1, prediction, average="micro")
```

Out[143]:
```
0.30612938555081387
```

In [154]:
```
inputs = Input(name='inputs',shape=[10000])
layer = Embedding(10000,50,input_length=10000)(inputs)
layer = LSTM(64)(layer)
layer = Dense(256,name='FC1')(layer)
layer = Activation('relu')(layer)
layer = Dropout(0.5)(layer)
layer = Dense(20,name='out_layer')(layer)
layer = Activation('sigmoid')(layer)
model = Model(inputs=inputs,outputs=layer)


plot_model(model, to_file='model1.png')
model.compile(loss='binary_crossentropy',optimizer=RMSprop(),metrics=['a
ccuracy'])
```

In [155]:
```
model.fit(xtrain_tfidf,y,batch_size=256,epochs=20,
          validation_split=0.2,callbacks=[EarlyStopping(monitor='val_los
s',min_delta=0.0001)])
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call l
ast)
<ipython-input-155-f02290bb9aa0> in <module>
      1 model.fit(xtrain_tfidf,y,batch_size=256,epochs=20,
----> 2           validation_split=0.2,callbacks=[EarlyStopping(monitor
='val_loss',min_delta=0.0001)])

~/opt/anaconda3/lib/python3.7/site-packages/keras/engine/training.py in
fit(self, x, y, batch_size, epochs, verbose, callbacks, validation_spli
t, validation_data, shuffle, class_weight, sample_weight, initial_epoc
h, steps_per_epoch, validation_steps, validation_freq, max_queue_size,
 workers, use_multiprocessing, **kwargs)
   1152             sample_weight=sample_weight,
   1153             class_weight=class_weight,
-> 1154             batch_size=batch_size)
   1155
   1156         # Prepare validation data.

~/opt/anaconda3/lib/python3.7/site-packages/keras/engine/training.py in
_standardize_user_data(self, x, y, sample_weight, class_weight, check_a
rray_lengths, batch_size)
    635             # Check that all arrays have the same length.
    636             if check_array_lengths:
--> 637                 training_utils.check_array_length_consistency(x
, y, sample_weights)
    638             if self._is_graph_network:
    639                 # Additional checks to avoid users mistakenly

~/opt/anaconda3/lib/python3.7/site-packages/keras/engine/training_util
s.py in check_array_length_consistency(inputs, targets, weights)
    242                          'the same number of samples as target
 arrays. '
    243                          'Found ' + str(list(set_x)[0]) + ' inp
ut samples '
--> 244                          'and ' + str(list(set_y)[0]) + ' targe
t samples.')
    245     if len(set_w) > 1:
    246         raise ValueError('All sample_weight arrays should have
 '

ValueError: Input arrays should have the same number of samples as targ
et arrays. Found 33830 input samples and 42288 target samples.
```

Text Processing functions

In [87]:
```
"""
Tokenize a text by double line breaks
"""
def line_break_tokenizer(input_text:str):
    # Divide doc input by double line break
    return input_text.split('\n\n')
"""
Tokenize a list of words into paragraph of count size
"""
def word_count_tokenizer(word_col, count):
    # Divide word_list input by double line break
    result = []
    for i in range(0, len(word_col), count):
        result.append(' '.join(word_col[i:i + count]))

    return result
"""
Use regex to remove punctuation, numbers and multi spaces
"""
def clean_text(text:str):
    clean = re.sub('[\W_]+', ' ', text.lower())
    clean = re.sub('[\d]+', ' ', clean)

    return re.sub(' +', ' ', clean)

def remove_stopwords(text:str):
    stop_words = set(stopwords.words('english'))
    no_stopword = [w for w in text.split() if not w in stop_words]
    return ' '.join(no_stopword)

"""
Preprocess of text
"""
def preprocess(doc, label, sample_size):
    # Divide doc into multiple paragraphs by total word/sample_size
    all_word = clean_text(doc).split()
    paragraphs = word_count_tokenizer(all_word, math.ceil(len(all_word)/
sample_size))
    # Create the df with classification
    labels = np.ones((sample_size,)) * label
    df = pd.DataFrame({'paragraph': paragraphs, 'label': labels })

    return df, df.count() + 1
```

TF-IDF Calculation

In [14]:
```
def convert_to_mat(index):
    mat = np.zeros((index.size, index.max()+1))
    mat[np.arange(index.size),index] = 1

    return mat

def computeTF(word_list, doc_size):
    tfDict = []
    for i  in range(0, len(word_list)):
        dicts = {}
        for word, count in word_list[i].items():
            dicts[word] = count / float(doc_size)
        tfDict.append(dicts)

    return tfDict

def computeIDF(documents, final_word_list):
    N = len(documents)

    idfDict = dict.fromkeys(final_word_list, 0)
    for document in documents:
        for word, val in document.items():
            if val > 0:
                idfDict[word] += 1

    for word, val in idfDict.items():
        if(val != 0):
            idfDict[word] = float(math.log(float(N) / float(val)))
        else:
            idfDict[word] = 0

    return idfDict

def computeTFIDF(doc_word, idfs, key):
    tfidf = []
    for i  in range(len(doc_word)):
        dicts = {}
        for word, val in doc_word[i].items():
            dicts[word] = val * idfs[word]
        dicts['123'] = key
        tfidf.append(dicts)

    return tfidf
```

In [161]:
```
book_data = pd.read_csv("booksummaries.txt", delimiter="\t", )
book_data
```

Out[161]:

	Index	symbol	Title	Author	Publish	Genre	Plot
0	620	/m/0hhy	Animal Farm	George Orwell	1945-08-17	{"/m/016lj8": "Roman \u00e0 clef", "/m/06nbt":...	Old Major, the old boar on the Manor Farm, ca...
1	843	/m/0k36	A Clockwork Orange	Anthony Burgess	1962	{"/m/06n90": "Science Fiction", "/m/0l67h": "N...	Alex, a teenager living in near-future Englan...
2	986	/m/0ldx	The Plague	Albert Camus	1947	{"/m/02m4t": "Existentialism", "/m/02xlf": "Fi...	The text of The Plague is divided into five p...
3	1756	/m/0sww	An Enquiry Concerning Human Understanding	David Hume	NaN	NaN	The argument of the Enquiry proceeds by a ser...
4	2080	/m/0wkt	A Fire Upon the Deep	Vernor Vinge	NaN	{"/m/03lrw": "Hard science fiction", "/m/06n90...	The novel posits that space around the Milky ...
...	...	...	...	...	...	...	...
16554	36934824	/m/0m0p0hr	Under Wildwood	Colin Meloy	2012-09-25	NaN	Prue McKeel, having rescued her brother from ...
16555	37054020	/m/04f1nbs	Transfer of Power	Vince Flynn	2000-06-01	{"/m/01jfsb": "Thriller", "/m/02xlf": "Fiction"}	The reader first meets Rapp while he is doing...
16556	37122323	/m/0n5236t	Decoded	Jay-Z	2010-11-16	{"/m/0xdf": "Autobiography"}	The book follows very rough chronological ord...
16557	37132319	/m/0n4bqb1	America Again: Re-becoming The Greatness We Ne...	Stephen Colbert	2012-10-02	NaN	Colbert addresses topics including Wall Stree...
16558	37159503	/m/073nkd	Poor Folk	Fyodor Dostoyevsky	1846	{"/m/02ql9": "Epistolary novel", "/m/014dfn": ...	Makar Devushkin and Varvara Dobroselova are s...

16559 rows × 7 columns

In [162]:
```
book_data['clean_plot'] = book_data['Plot'].apply(lambda text: clean_tex
t(text))
book_data['clean_plot'] = book_data['clean_plot'].apply(lambda text: rem
ove_stopwords(text))
```

In [163]:
```
book_data
```

Out[163]:

	Index	symbol	Title	Author	Publish	Genre	Plot
0	620	/m/0hhy	Animal Farm	George Orwell	1945-08-17	{"/m/016lj8": "Roman \u00e0 clef", "/m/06nbt":...	Old Major, the old boar on the Manor Farm, ca...
1	843	/m/0k36	A Clockwork Orange	Anthony Burgess	1962	{"/m/06n90": "Science Fiction", "/m/0l67h": "N...	Alex, a teenager living in near-future Englan...
2	986	/m/0ldx	The Plague	Albert Camus	1947	{"/m/02m4t": "Existentialism", "/m/02xlf": "Fi...	The text of The Plague is divided into five p...
3	1756	/m/0sww	An Enquiry Concerning Human Understanding	David Hume	NaN	NaN	The argument of the Enquiry proceeds by a ser...
4	2080	/m/0wkt	A Fire Upon the Deep	Vernor Vinge	NaN	{"/m/03lrw": "Hard science fiction", "/m/06n90...	The novel posits that space around the Milky ...
...	...	...	...	...	...	...	...
16554	36934824	/m/0m0p0hr	Under Wildwood	Colin Meloy	2012-09-25	NaN	Prue McKeel, having rescued her brother from ...
16555	37054020	/m/04f1nbs	Transfer of Power	Vince Flynn	2000-06-01	{"/m/01jfsb": "Thriller", "/m/02xlf": "Fiction"}	The reader first meets Rapp while he is doing...
16556	37122323	/m/0n5236t	Decoded	Jay-Z	2010-11-16	{"/m/0xdf": "Autobiography"}	The book follows very rough chronological ord...
16557	37132319	/m/0n4bqb1	America Again: Re-becoming The Greatness We Ne...	Stephen Colbert	2012-10-02	NaN	Colbert addresses topics including Wall Stree...
16558	37159503	/m/073nkd	Poor Folk	Fyodor Dostoyevsky	1846	{"/m/02ql9": "Epistolary novel", "/m/014dfn": ...	Makar Devushkin and Varvara Dobroselova are s...

16559 rows × 8 columns

In [175]:
```
book_tfidf = tfidf_vectorizer.transform(book_data['clean_plot'])

prediction = model.predict(book_tfidf > .14).astype(int)

r = multilabel_binarizer.inverse_transform(prediction)

r[0]
```

Out[175]:
```
('Animation', 'Family')
```

In [179]:
```
pd.set_option('display.max_colwidth', None)
book_data[book_data['Index']==48648].Genre
```

Out[179]:
```
158    {"/m/0dwly": "Children's literature", "/m/01hmnh": "Fantasy", "/
m/014dfn": "Speculative fiction", "/m/02xlf": "Fiction"}
Name: Genre, dtype: object
```

In [176]:
```
r[158]
```

Out[176]:
```
('Adventure', 'Family')
```

In [ ]: