TED Talks Recommendation System and Summarization

DATA SCIENCE INTENSIVE CAPSTONE PROJECT

ROJA RANI JYOTHI

11/30/2020

Table of Contents

Abstract

TED Talks are giving so much information and knowledge about fields that are alien to the audience and amaze them in the form of outstanding short stories, breathtaking visuals and subtle humor. We analyzed the datasets containing transcripts available in www.kaggle.com of all audio-video recordings of TED Talks uploaded to the official TED.com website until September 21st, 2017. By analyzing the datasets, we calculated similarity between TED Talks and based on the similarity, we built a recommendation system to decide which talk they might watch. And we did TED transcript summarization using spaCy, Gensim, LSA and LDA. Further, we did topic modelling on the Transcript dataset using NMF (Non-negative Matrix Factorization) around transcript data to cluster the TED talks. Finally, we built a predictive model which could predict the rating of a TED talk. Our model with tuning of parameters performs quite well with an accuracy score of 55% and F1(weighted) score of 50% using the Random Forest machine learning model.

1. Introduction

TED, which operates under the slogan 'Ideas worth spreading' is a nonprofit devoted to spreading ideas, usually in the form of short, powerful talks (18 minutes or less). TED began in 1984 as a conference where Technology, Entertainment and Design converged, and today covers almost all topics from science to business to global issues in more than 100 languages. The first TED, conceived by Richard Saul Wurman, who co-founded with Harry Marks, included a demo of the compact disc, the e-book and cutting-edge 3D graphics from Lucasfilm, while mathematician Benoit Mandelbrot demonstrated how to map coastlines using his developing theory of fractal geometry. But despite a stellar lineup, the event lost money, and it was six years before Wurman and Marks tried again.

The TED Conference became an annual event in Monterey, California, attracting a growing and influential audience from many different disciplines united by their curiosity and open-mindedness and also by their shared discovery of an exciting secret. It has been held since 1990. Back then, TED was an invitation only event. It is now you are welcome and encouraged to apply to attend.

Goals and Possible questions that we accomplished from the available datasets

Our main goal here in this project is to find out the similarity between TED Talks and based on those similarities to build a recommendation system. And second one is TED Talk summarization. At the same time, our focus would be to answer the below questions from the dataset available to get meaningful derivations and inferences.

1. Which themes are most popular among TEDsters?
2. Which are the most viewed and most favorited Talks of all time?
3. What kind of topics attract the maximum discussion and debate (in the form of comments)?
4. TED Talk which has more influence on the crowd taking the number of views into consideration?
5. Number of languages TED talk is released in, number of views, number of responses received over them?
6. Occupation of the speaker and how it is related to TED talks?
7. Rating of TED Talk and which TED Talks are getting the most Inspirational rating?
8. Which months are most popular among TED chapters?

Approach:

To examine the specific problem, we will apply a full Data Science life cycle and NLP techniques composed of the following steps:

1. Data Wrangling to audit the quality of the data and perform all the necessary actions to clean the dataset.
2. Data Exploration for understanding the variables and creating intuition on the data.
3. Data preprocessing which includes text preprocessing. Using NLP techniques we will clean the text data and create word vectorizer for text data.
4. Data Normalization and Data Transformation for preparing the dataset for the learning algorithms (if needed).
5. Training / Testing data creation to evaluate the performance of our models and fine-tune their hyperparameters.
6. Model selection and evaluation. This will be the final model for our goal.

Target Clients

These meaningful derivation, inference and results can be used by several kinds of people with different aims in mind.

- Audience who wants to know the summary of the talk from a specific area without further going through the whole talk.

- Audience who are looking for the content relevant contextual recommendations.
- Another audience of our result can be a person looking to gain information from any TED talks from a specific area. Our result would help such people to know which talk and speaker has the most popularity in that area as a suggestive guide. It is highly likely that a popular talk will have some informative message and learnings thus the listener can be benefited most.
- A TED talk presenter can use it to know the common topic in which people are more interested to listen and thus he can plan the topic of his talk accordingly.
- Also, the patterns from rating data for a TED talk can help speakers to know whether their talk has been informative or confusing or ok and improve accordingly.

2. Data Wrangling

2.1 Data Collection

The source of the dataset contains information about all audio-video recordings of TED Talks uploaded to the official TED.com website until September 21st, 2017. The TED main dataset contains information about all talks including number of views, number of comments, descriptions, speakers and titles. The TED transcripts dataset contains the transcripts for all talks available on TED.com.

https://www.kaggle.com/rounakbanik/ted-talks

This dataset contains two csv files:

ted_main.csv - Contains data on actual TED Talk metadata and TED Talk speakers.
transcripts.csv - Contains transcript and URL information for TED Talks.

In this part we merged the two datasets as ted_talks dataset.

2.2 Data Definition

Columns description for the ted_main.csv file

Columns	Description
name	The official name of the TED Talk. Includes the title and the speaker
title	The title of the talk
description	A blurb of what the talk is about

main_speaker	The first named speaker of the talk
speaker_occupation	The occupation of the main speaker
num_speaker	The number of speakers in the talk
duration	The duration of the talk in seconds
views	The number of views on the talk
event	The TED/TEDx event where the talk took place
film_date	The Unix timestamp of the filming
published_date	The Unix timestamp for the publication of the talk on TED.
comments	The number of first level comments made on the talk
tags	The themes associated with the talk
languages	The number of languages in which the talk is available
ratings	A stringified dictionary of the various ratings given to the talk (inspiring, fascinating, jaw dropping, etc.)
related_talks	A list of dictionaries of recommended talks to watch next
url	The URL of the talk

Columns description for the transcript.csv file

transcript	Transcript of the talk
url	The URL of the talk

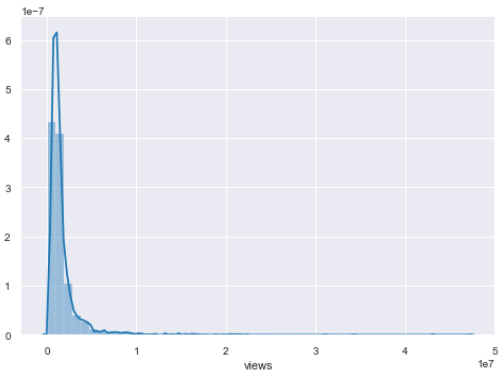
2.3 Data Cleaning

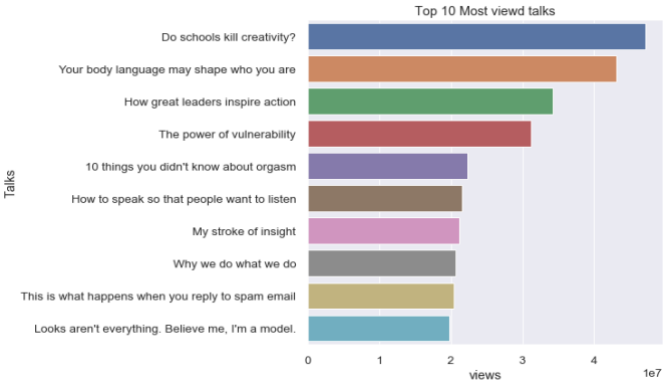
- Firstly, I rearranged the column names in the appropriate order for convenience.
- Next, I checked the NULL value presence and found 6 in speaker_occupation and 86 in the transcript column. And we did not remove NULL values at this stage for further data analysis.
- Typecasting the columns - Date present in the dataset is in Unix format so we processed Unix Date and changed it to YYYY-MM-DD format and created day, month and year columns for further analysis.
- Changed the duration column values from seconds to minutes.
- Checked the duplicates and removed them.

3. Exploratory Data Analysis

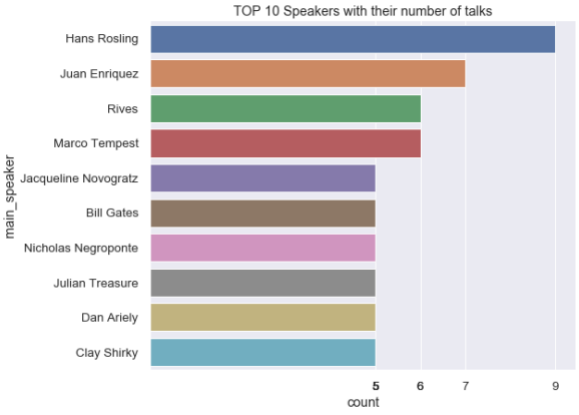
- First, I checked the outliers and did not remove them at this stage as those are the real values for our analysis.
- I plot the heat map to check the correlation between features.

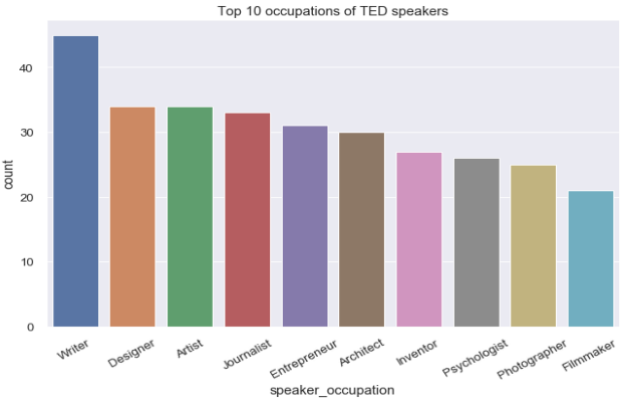
- We analyzed the distribution of views and observed that the majority of talks have views less than 5 million. Then we investigated the most viewed talks of all time and witnessed that the Talk **Do schools kill creativity?** is the most viewed talks of all times with more than 40 million views.



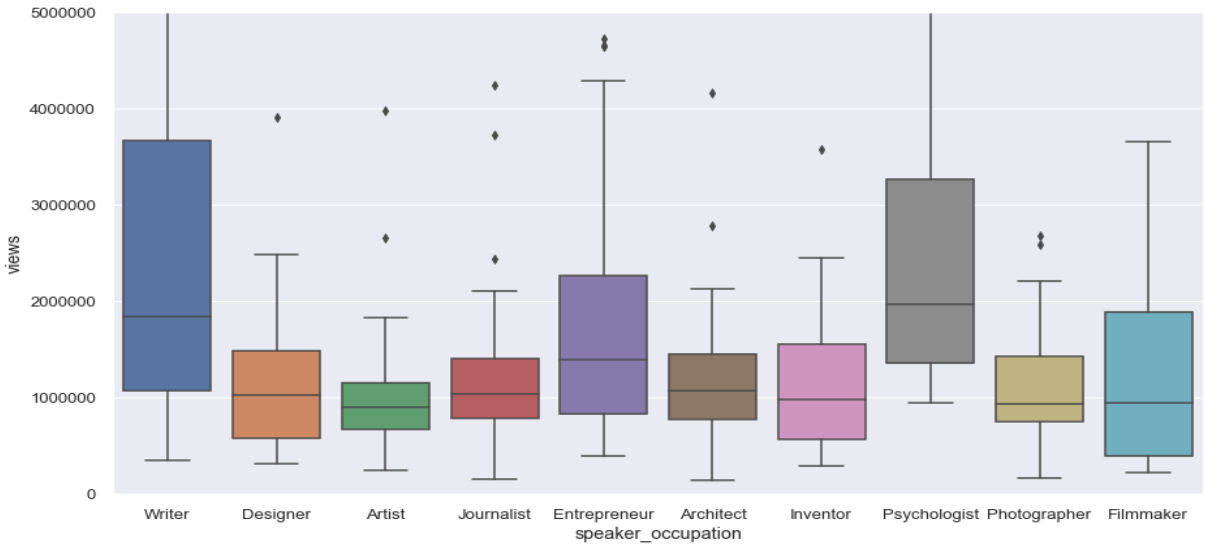


- Next, we analyzed which speaker has given the most number of TED Talks, that is who is the most popular speaker with respect to the number of Talks and their occupation. Summary findings that **Hans Rosling** is the most popular TED Speaker with 9 talks on the TED platform and **Writers** are the most popular that TED is interested in inviting to its events.





- Now we will analyze which occupation tends to gather more views and try to figure it out. Observation results that on an average **Psychologist** occupation tend to accumulate more views out of top 10 most popular occupations.



- Next, I tried to figure out is there any relationship between speaker_occupation and the views? For that I made a hypothesis to prove it statistically.

HYPOTHESIS

- Null Hypothesis(H0) - There is no significant difference among different occupation groups.
- Alternate Hypothesis (Ha) - There is a significant difference among different occupation groups.

ANOVA Test:

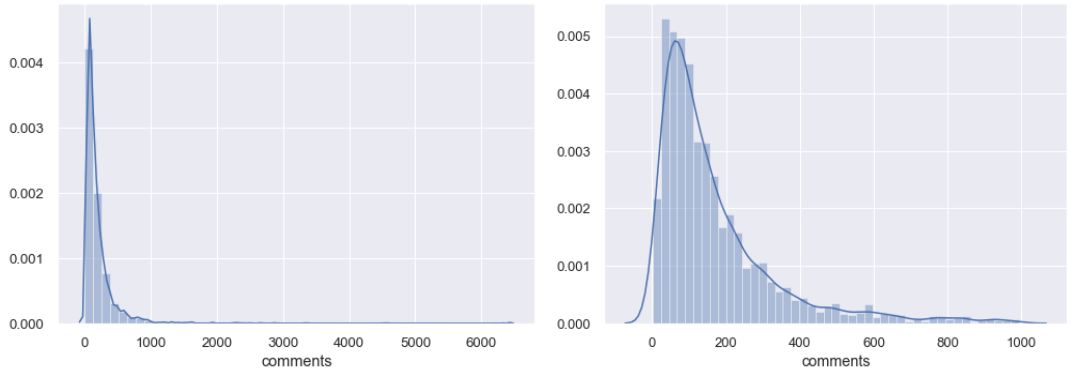
- ANOVA is used when we want to compare the means of a condition between more than two groups. It tests if there is a difference in the mean somewhere in the model (testing if there was an overall effect).
- There are 3 types of ANOVA test. Here we are going to use One-Way ANOVA(A one-way ANOVA has just one independent variable).There are 2 ways to perform One-Way ANOVA.
    - 1.One-Way ANOVA Test using stats model's module
    - 2.One-Way ANOVA Test using OLS Model
- Here we used One-Way ANOVA Test using the OLS Model. And the results for overall model F (1458, 1091) = 0.859, p = 0.9965.

- From the above results:

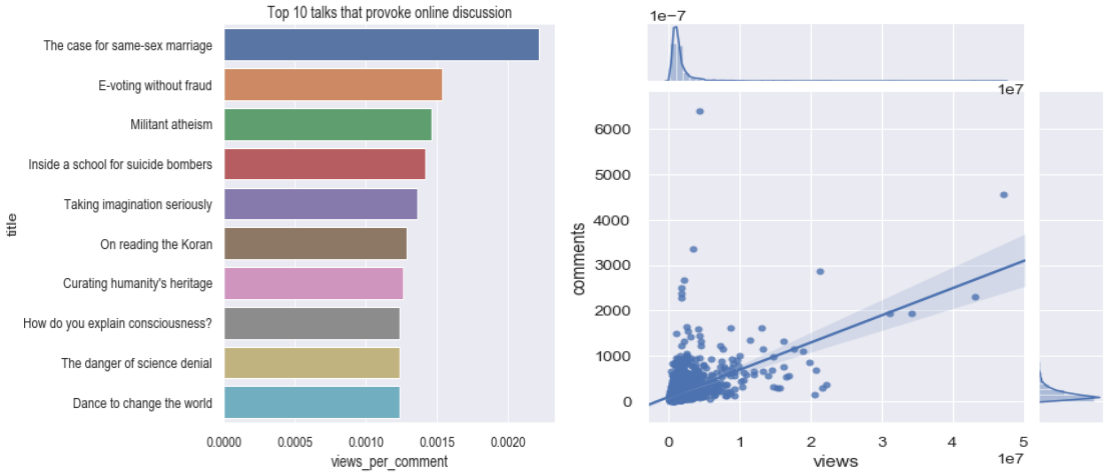
We see that the p-value is greater than 0.05. Therefore, we failed to reject the Null Hypothesis that there's no difference among different occupation groups. So we

concluded that **There is no statistical significant relationship between views and speakers occupation**.

- Now I analyzed the distribution of views and found that the mean is highly affected by the outliers. And I took a cutoff point of 1000 for in depth analysis.

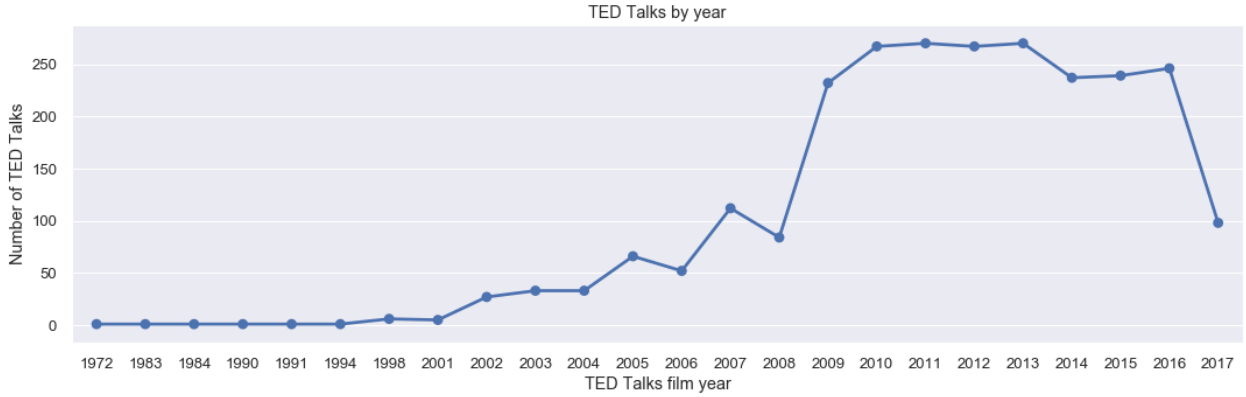


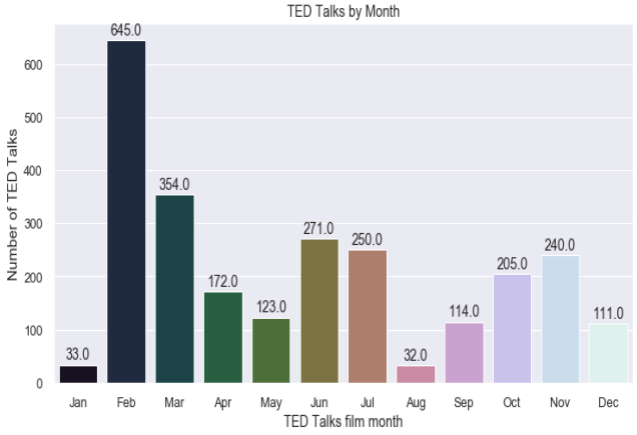
- Next, I analyzed how many comments the talk gathered depends on its views. That is **which talks provoke the most online discussion?** Observation results that the **case for same_sex marriage** is the most hotly debated talk of all time. And then I made some hypotheses to figure out that Is there any relationship between views and comments? Pearson Coefficient statistical test proved that there is a relationship between views and comments.

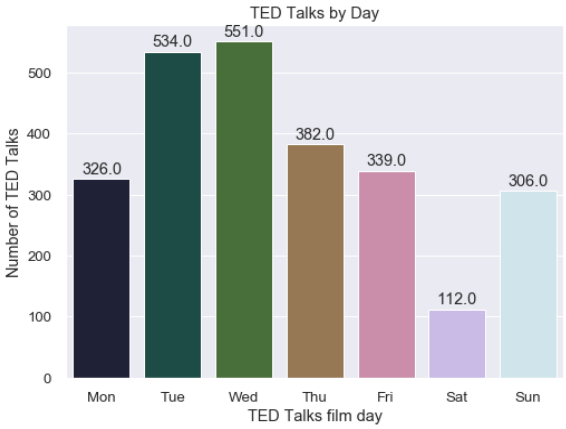


- Next I analyzed the TED Talks by year, month and day. Observation results showed that the number of talks increased gradually over the years, the month February is the most popular

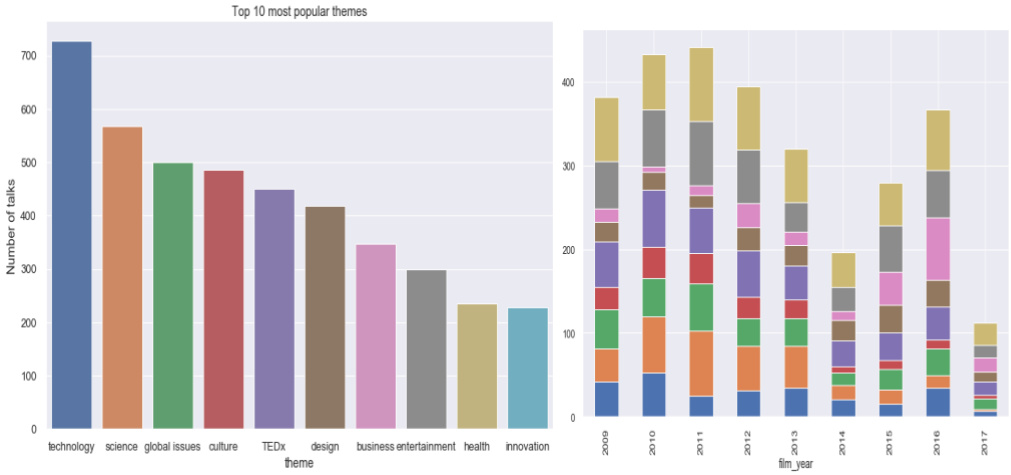
month for TED conference and surprisingly, Wednesday and Tuesday are the most popular days for TED conferences instead of weekends.

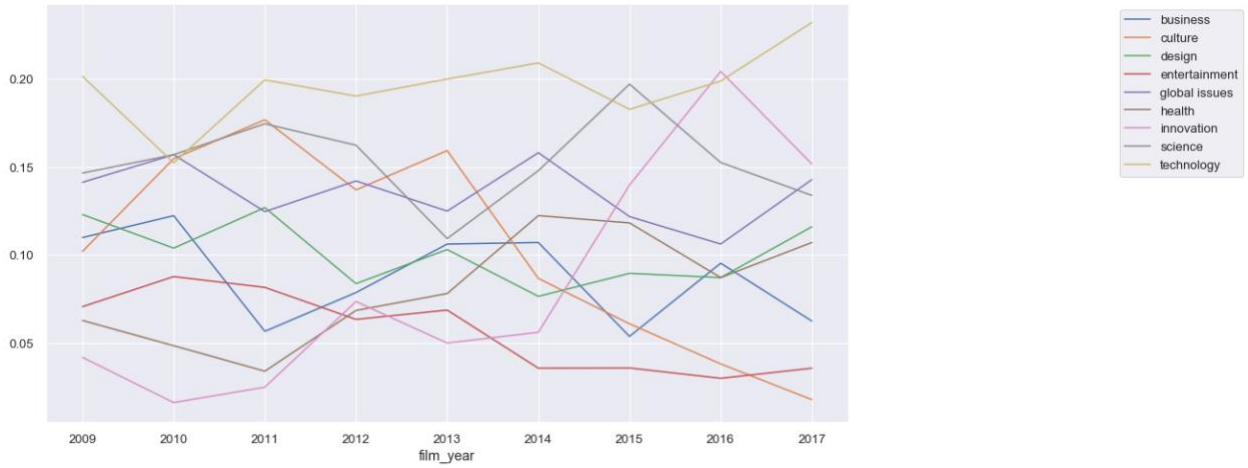


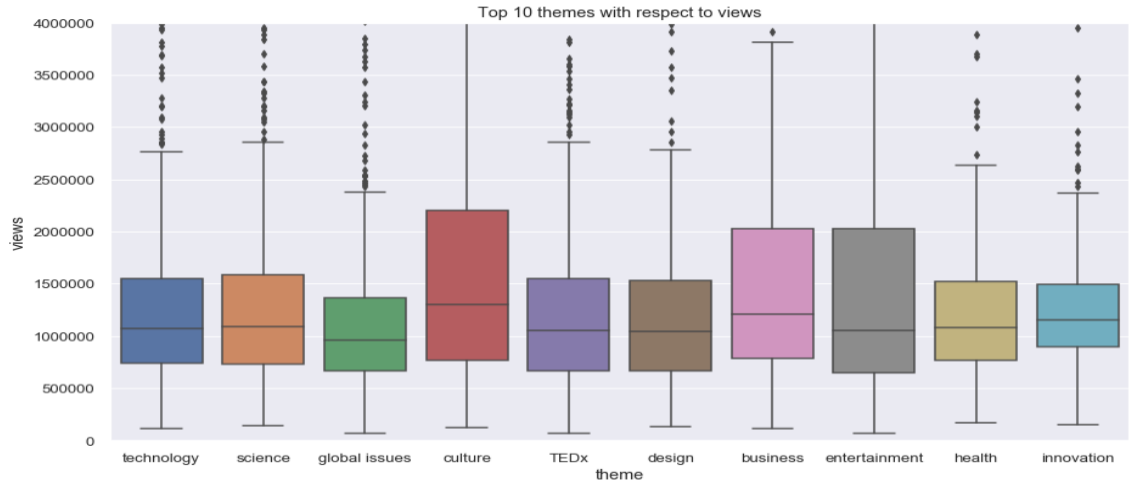




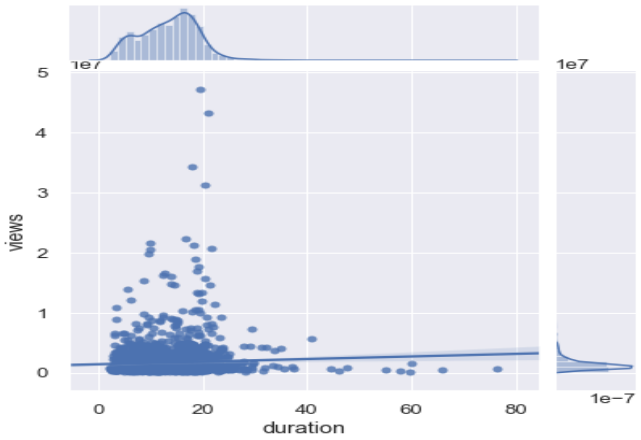
- Next, I analyzed the TED themes and plotted the top 10 themes by year. Observation tells that Technology is the most popular theme among 416 different categories of themes and has steadily increased over the years with a slight dip in 2010 and has been the most in 2017.And we also analyzed if certain themes tend to gather more views than other themes. Observation says that Culture has gathered more median number of views followed by business and innovation.


Top 10 most popular themes
Number of talks
0
100
200
300
400
500
600
700
technology
science
global issues
culture
TEDx
design
business
entertainment
health
innovation
theme
0
100
200
300
400
2009
2010
2011
2012
2013
2014
2015
2016
2017
film_year
business
culture
design
entertainment
global issues
health
innovation
science
technology

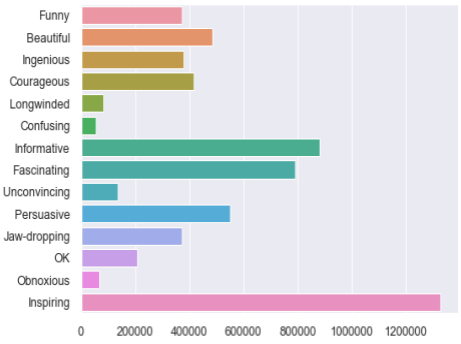

0.05
0.10
0.15
0.20
2009
2010
2011
2012
2013
2014
2015
2016
2017
film_year
business
culture
design
entertainment
global issues
health
innovation
science
technology

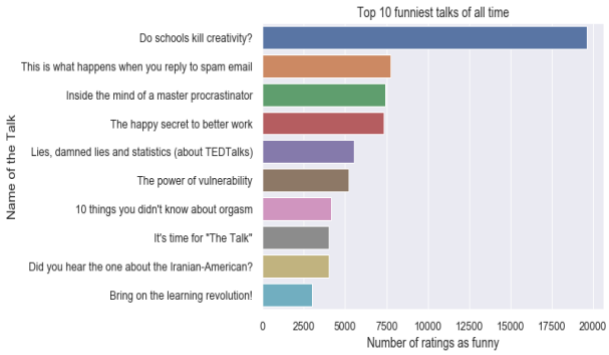

Top 10 themes with respect to views
views
0
500000
1000000
1500000
2000000
2500000
3000000
3500000
4000000
technology
science
global issues
culture
TEDx
design
business
entertainment
health
innovation
theme

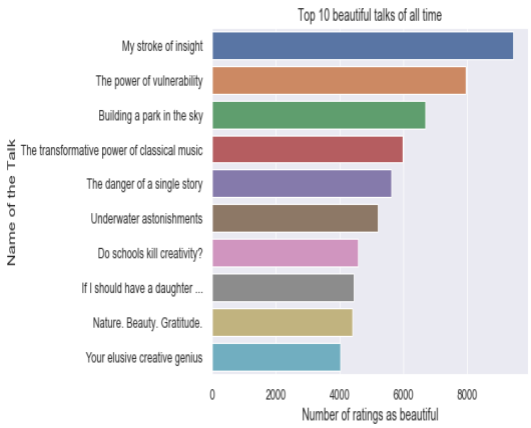
- Next, I analyzed the TED Talks duration and witnessed that on an average Ted talks are 13.78 minutes long. I also explored the relationship between views and duration to check the length of the talk that accumulates the views. Summary results that there is no statistical significance relationship between the length of the talks and its views.

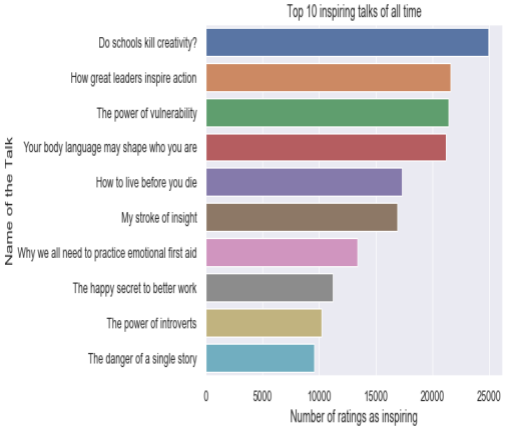


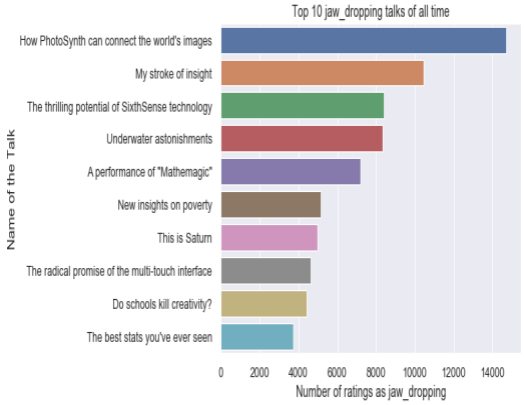
- Next, I analyzed the ratings of TED Talks and witnessed that on average, Ted talks are getting 2436.41 ratings per each talk and Most of the TED Talks are getting **Inspiring** ratings. And also visualized the top 10 most Funniest, Beautiful, Inspiring, Jaw-dropping, Confusing TED Talks.

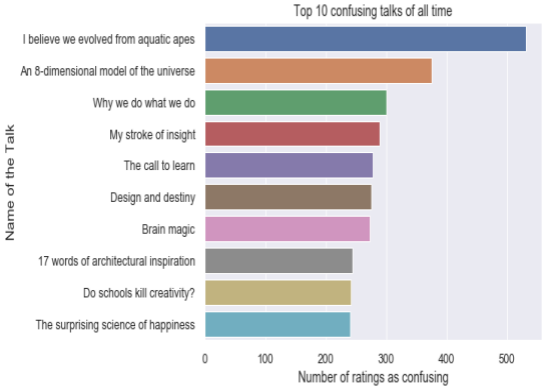












- Next, I investigated the TED word cloud to know about which words are most often used by TED Speakers. Summary findings that the word **one** is the popular word that the TED speakers often used in their talks.

- Also I visualized the TED themes and TED speakers occupation word clouds and witnessed that most of the TED Talks are about **technology** themes and most of the TED Speakers' profession is **Writer**.

- At last, I analyzed the relationship between numerical features using correlation matrix to select and remove collinear features (features that are more than 95% correlated for removal).

EDA Conclusion:

We have discovered so many interesting insights and trends in data through exploratory data analysis. We discovered the most viewed TED Talks of all time, analyzed which speaker has given the most number of TED Talks. We also analyzed the speakers occupation and visualized the speakers occupation with respect to views and made hypotheses to figure out is there any relationship between occupation and views. Analyzed the TED Talks comments, visualized the comets with respect to views and how many comments the talk gathered depends on its views. Analyzed the TED Talks languages and the relationship between language and views. Visualized the TED Talks by year, month and day. Analyzed the top 10 most TED Talk themes, visualized themes by year and themes with respect to views. Analyzed the TED Talks duration and their relationship with views. And we also analyzed the TED Talks ratings and visualized the top 10 most funniest, inspiring, beautiful, jaw-dropping and confusing talks. We investigated the TED word cloud to know about which words are most often used by TED Speakers. And we also visualized the TED Talks themes and speaker's occupation word cloud. Even though, we can do a lot more interesting analysis and can draw more meaningful insights, but I ended up my analysis here and moved on to the preprocessing and modeling step.

4. Data Preprocessing

4.1 Language detection:

In this step, with the langdetect package I detected the language of the TED transcript. Observation results that there are 2463 transcripts in english,1 transcript in french and rest of them are NULL transcripts. I deleted the NULL and french transcript to make sure I am dealing with the same language.

4.2 Text Preprocessing:

Data preprocessing is the phase of preparing raw data to make it suitable for a machine learning model. For NLP, that includes text cleaning, stopwords removal, stemming and lemmatization.

Text cleaning steps vary according to the type of data and the required task. Generally, the string is converted to lowercase and punctuation is removed before text gets tokenized. Tokenization is the process of splitting a string into a list of strings (or "tokens").

By using spaCy library, we did text normalization which includes:

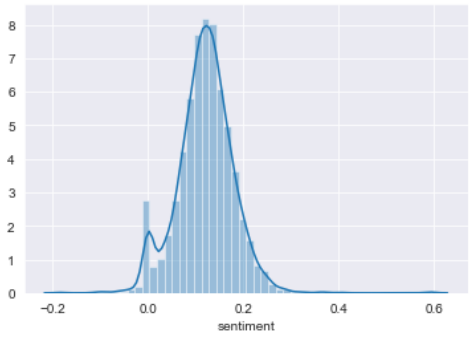
- converting all letters to lower or upper case
- converting numbers into words or removing numbers
- removing punctuations, accent marks and other diacritics
- removing white spaces
- removing the words which are in parenthesis

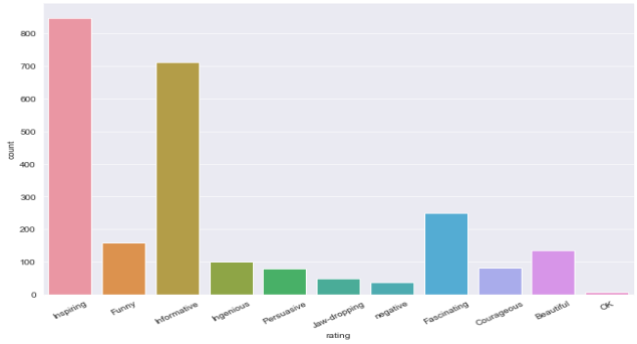
- expanding abbreviations
- removing stop words, sparse terms, and particular words

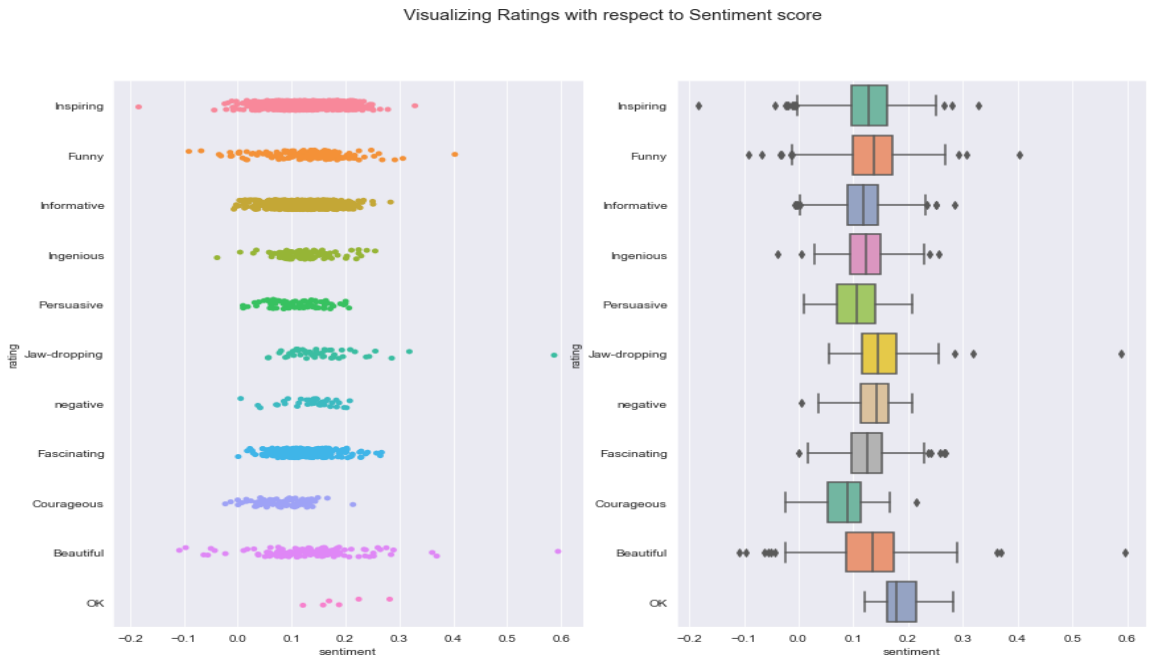
4.3 Feature Extraction:

- In this step we added extra features that are extracted from length analysis. It's important to have a look at the length of the text because it's an easy calculation that can give a lot of insights. Maybe, for instance, we are lucky enough to discover that one category is systematically longer than another and the length would simply be the only feature needed to build the model. We extracted word_count, char_count, sentence_count, avg_word_length, avg_sentence_length feature through length analysis. And we did sentiment analysis on TED transcripts using the Textblob library and extracted the feature as sentiment. Sentiment analysis is the representation of subjective emotions of text data through numbers or classes. Textblob, built on top of NLTK, is one of the most popular libraries, it can assign polarity to words and estimate the sentiment of the whole text as an average. The polarity score is a float within the range [-1.0, 1.0]. And we visualized the distribution of sentiment of the transcript.

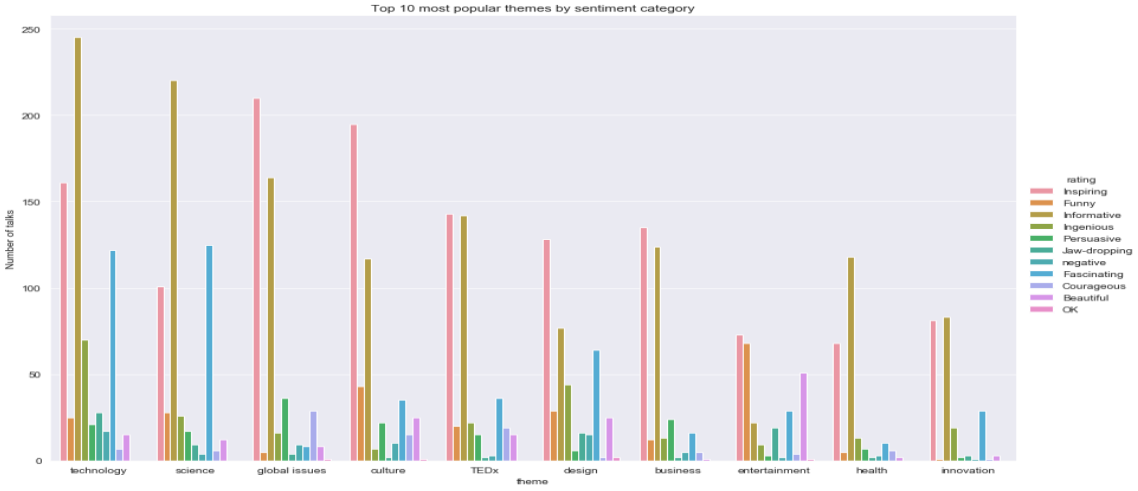
- Rating column includes the keyword and its associated hit indicating the emotion of the viewer. For example, the first TED talk has a count of 19645 for 'Funny' keyword, which may mean that the viewer found this talk a lot more funny than 'Obnoxious' which has a count of 209. So, considering these keywords to provide us some useful information about viewer's sentiment about the TED talk, we first took out all the possible keywords we have in the current TED talk data. We found 14 such keywords, same across the entire TED talks 'rating' column which are ingenious, Informative, Funny, Confusing, Jaw-dropping, Inspiring, Fascinating, Unconvincing, Persuasive, OK, Obnoxious, Long Winded, and Beautiful. And we derived the rating columns with the emotion which got the highest number of comments among those 14 emotions. And then we visualized the ratings columns and ratings with respect to sentiment score of transcript.







- And we also visualized the top 10 most popular themes by rating category and summary findings that **technology, Science, health, innovation** themes of TED Talks got Informative as the primary ratings.



-

4.4 Feature Engineering:

In this step we extracted features from existing features like from ratings feature we extracted Ingenious, Informative, Funny, Confusing, Jaw-dropping, Inspiring, Fascinating, Unconvincing, Persuasive, OK, Obnoxious, Long Winded, and Beautiful features.

5. TED Talk Summarization:

In this step we did TED Talk summarization using different algorithms, but we found the talk summarization using spaCy library makes sense. Text summarization can broadly be divided into two categories

- **Extractive Summarization:**

  These methods rely on extracting several parts, such as phrases and sentences, from a piece of text and stack them together to create a summary. Therefore, identifying the right sentences for summarization is of utmost importance in an extractive method.

- **Abstractive Summarization**:

  These methods use advanced NLP techniques to generate an entirely new summary. Some parts of this summary may not even appear in the original text.

Here I focused on the extractive summarization technique. And the algorithms we tried are

5.1 Text summarization using spaCy

- spaCy is a free, open-source advanced natural language processing library, written in the programming languages Python and Cython. spaCy is mainly used in the development of production software and also supports deep learning workflow via statistical models of PyTorch and TensorFlow.
- spaCy provides a fast and accurate syntactic analysis, named entity recognition and ready access to word vectors. We can use the default word vectors or replace them with any you have. spaCy also offers tokenization, sentence boundary detection, POS tagging, syntactic parsing, integrated word vectors, and alignment into the original string with high accuracy.
- In this step we built a function with the following steps which extract the summary of the talk
    1. Filtering the tokens depends on the POS tagging
    2. Normalizing the tokens
    3. Weighing the sentences
    4. Depending on the sentence weight, we are going to take the top 3 most sentences and join them to form a summary

In [656]:
```
#checking the summaraiztion for 2451th transcript in the dataset as an
text_summary_spacy(ted_talks['transcript'][2451])
```

Out[656]:
```
'All life on Earth requires water, so in my case I focus on the intimate
relationship between water and life in order to understand if we could fi
nd life in a planet as dry as Mars. So I remembered that I usually see fo
gs in Yungay, so after setting sensors in a number of places, where I rem
ember never seeing fogs or clouds, I reported four other sites much drier
than Yungay, with this one, María Elena South, being the truly driest pla
ce on Earth, as dry as Mars, and amazingly, just a 15-minute ride from th
e small mining town where I was born. Now, in this search, we were trying
to actually find the dry limit for life on Earth, a place so dry that not
hing was able to survive in it.'
```

5.2 Text Summarization using Gensim with TextRank

- **gensim** is a very handy python library for performing NLP tasks. The text summarization process using the gensim library is based on the TextRank Algorithm.
- **TextRank** is an extractive summarization technique. It is based on the concept that words which occur more frequently are significant. Hence, the sentences containing highly frequent words are important.

Based on this, the algorithm assigns scores to each sentence in the text. The top-ranked sentences make it to the summary. The default parameters of the **summarize** function are

ratio: It can take values between 0 to 1. It represents the proportion of the summary compared to the original text.

word_count: It decides the no of words in the summary.

In this step we imported the library summarize from gensim.summarization and summarized the ted talk by taking the word count of 100 into account.

In [648]:
```
from gensim.summarization import summarize
text = re.sub(r"\((.*?)\)|—|\"|\n|\+", r" ", ted_talks['transcript'][24
text = " ".join(text.split())
summarize(text,word_count = 100)
```

Out[648]:
```
"In this place, we reported a new type of microalgae that grew only on to
p of the spiderwebs that covered the cave entrance.\nIt's covered with de
w, so this microalgae learned that in order to carry photosynthesis in th
e coast of the driest desert on Earth, they could use the spiderwebs.\nTh
ese type of findings suggest to me that on Mars, we may find even photosy
nthetic life inside caves.\nBut even here, well hidden underground, we fo
und a number of different microorganisms, which suggested to me that simi
larly dry places, like Mars, may be in inhabited."
```

5.3 Text Summarization with Sumy

sumy library provides you several algorithms to implement Text Summarization. We implemented the below algorithms for summarization using sumy:

- LexRank
- Luhn
- Latent Semantic Analysis, LSA
- KL-Sum

Latent Semantic Analysis (LSA)

- Latent Semantic Analysis is an unsupervised learning algorithm that can be used for extractive text summarization.
- It extracts semantically significant sentences by applying singular value decomposition (SVD) to the matrix of term-document frequency.
- In this step we imported the LsaSummarizer library from sumy.summarizers.lsa and did summarization by taking the account of 3 sentences.

Likewise we tried the other algorithms by importing the KLSummarizer, LexRankSummarizer, LuhnSummarizer from sumy.summarizers and extracted the summary of TED Talk by taking the 3 sentences into account.

From all the above algorithms, we found that **summarization with spaCy** makes sense to me. So I used spaCy to summarize all the TED Talks.

6. TED Talk Recommendation System:

The idea is to demonstrate how one can generate recommendations just using content. This becomes essentially important when you don't have any user-item interaction data, essentially when you are starting out new and still want to provide the consumers of your content relevant contextual recommendations.

We did recommendation system by following these steps:

- Text Preprocessing
- Tfidf Vectorizer
- Generate Cosine similarity matrix
- Built a recommendation function based on similarities.

In this step first we loaded the preprocessed TED Talks dataset and created word vectorizer using Tfidf Vectorizer on transcript. And then calculated the cosine similarity to find out how similar the TED Talks to each other. Next, we build a recommender function based on cosine similarity to get the top 10 most similar talks.

Term Frequency-Inverse Document Frequency (Tf-Idf):

Term Frequency measures how often the word appears in a given document, while Inverse document frequency measures how rare the word is in a corpus. The product of these two quantities, measures the importance of the word and is known as Tf-Idf.

Finding similar TED Talks

- To find out similar talks among different talks, we will need to compute a measure of similarity. Usually when dealing with Tf-Idf vectors, we use cosine similarity.
- The cosine similarity will become a means for us to find out how similar the transcript of one Ted Talk is to the other.

In [21]:
```
%%time
from sklearn.metrics.pairwise import cosine_similarity
cosine_sim = cosine_similarity(tfidf_matrix)
```

```
CPU times: user 1.23 s, sys: 68.6 ms, total: 1.3 s
Wall time: 1.31 s
```

Recommender function

- Firstly, we generated a mapping between titles and index. Next, we followed the below steps to build a function.

  STEPS

  - 1.Take a movie title, cosine similarity matrix, and indices series as arguments.
  - 2.Extract pairwise cosine similarity scores for the movie
  - 3.Sort the scores in descending order
  - 4.Output titles corresponding to the highest scores.

After generating the recommendation function, the output results the top 10 most similar talks along with their url link to the talk.

```
n [27]:  1 get_recommendations('Do schools kill creativity?',cosine_sim,indices)

ut[27]: (1413                How to escape education's death valley
         661                      Bring on the learning revolution!
         1958     How I stopped the Taliban from shutting down m...
         1823           How to run a company with (almost) no rules
         683                      Education innovation in the slums
         2224     How America's public schools keep kids in poverty
         372                       Mosquitos, malaria and education
         1411               Our failing schools. Enough is enough!
         1335                                  Kids need structure
         1364                            A girl who demanded school
         Name: title, dtype: object,
         1413    https://www.ted.com/talks/ken_robinson_how_to_... (https://www.t
        ed.com/talks/ken_robinson_how_to_...)
         661     https://www.ted.com/talks/sir_ken_robinson_bri... (https://www.t
        ed.com/talks/sir_ken_robinson_bri...)
         1958    https://www.ted.com/talks/sakena_yacoobi_how_i... (https://www.t
        ed.com/talks/sakena_yacoobi_how_i...)
         1823    https://www.ted.com/talks/ricardo_semler_how_t... (https://www.t
        ed.com/talks/ricardo_semler_how_t...)
         683     https://www.ted.com/talks/charles_leadbeater_o... (https://www.t
        ed.com/talks/charles_leadbeater_o...)
         2224    https://www.ted.com/talks/kandice_sumner_how_a... (https://www.t
```

7. TED Talk Topic Modeling:

In this step we tried two algorithms LDA (Latent Dirichlet Allocation) and NMF (Non-negative Matrix Factorization) for topic modeling

- LDA is based on probabilistic graphical modeling while NMF relies on linear algebra. Both algorithms take as input a bag of words matrix (i.e., each document represented as a row, with each column containing the count of words in the corpus). The aim of each algorithm is then to produce 2 smaller matrices, a document to topic matrix and a word to topic matrix that when multiplied together reproduce the bag of words matrix with the lowest error.
- Both NMF and LDA are not able to automatically determine the number of topics and this must be specified.

Latent Dirichlet Allocation (LDA)

Topic modeling is a type of statistical modeling for discovering the abstract “topics” that occur in a collection of documents. Latent Dirichlet Allocation (LDA) is an example of topic model and is used to classify text in a document to a particular topic. It builds a topic per document model and words per topic model, modeled as Dirichlet distributions.

We tried and tested the LDA with different parameters like number of topics and maximum number of features used and found that below result makes sense with number of topics set to 20 and maximum number of users set to 5000.The output results 20 topics, and some of them have noisy data and did not get any valid topic.

```
Topic 0:
['technology', 'like', 'want', 'come', 'know', 'thing', 'people', 'way',
'time', 'think']
Topic 1:
['da', 'humor', 'ha', 'da da', 'feminist', 'sensitive', 'thank thank', 's
pread', 'approve', 'gray']
Topic 2:
['people', 'think', 'human', 'way', 'like', 'thing', 'right', 'idea', 'qu
estion', 'world']
Topic 3:
['black', 'community', 'white', 'race', 'american', 'african', 'america',
'man', 'drug', 'color']
Topic 4:
['life', 'know', 'people', 'come', 'story', 'time', 'like', 'feel', 'da
y', 'tell']
Topic 5:
['crispr', 'technology', 'drown', 'broken', 'excitement', 'fee', 'pause',
'venture', 'delight', 'spark']
Topic 6:
['jihad', 'mid', 'like use', 'optimism', 'deposit', 'relax', 'optimisti
c', 'bin', 'question question', 'lot different']
Topic 7:
['city', 'use', 'design', 'new', 'build', 'like', 'work', 'create', 'buil
ding', 'project']
Topic 8:
```

Next, we tried the other method NMF with different parameters.

Non-Negative Matrix Factorization (NMF)

Non-Negative Matrix Factorization is a statistical method to reduce the dimension of the input corpora. It uses factor analysis method to provide comparatively less weightage to the words with less coherence. The below output results are way better than the LDA model.

```
0 :  ['war', 'government', 'political', 'democracy', 'violence', 'conflic
t', 'election', 'police', 'citizen', 'security']
1 :  ['patient', 'health', 'doctor', 'disease', 'medical', 'hospital', 'd
rug', 'medicine', 'surgery', 'treatment']
2 :  ['planet', 'earth', 'mars', 'solar', 'energy', 'atmosphere', 'star',
'sun', 'fly', 'ice']
3 :  ['business', 'economy', 'economic', 'market', 'growth', 'china', 'gl
obal', 'india', 'cost', 'government']
4 :  ['datum', 'computer', 'machine', 'internet', 'phone', 'web', 'digita
l', 'video', 'online', 'algorithm']
5 :  ['brain', 'neuron', 'memory', 'cortex', 'disorder', 'behavior', 'ani
mal', 'consciousness', 'signal', 'pattern']
6 :  ['robot', 'robotic', 'machine', 'leg', 'intelligence', 'sensor', 'ar
tificial', 'interact', 'video', 'autonomous']
7 :  ['universe', 'galaxy', 'particle', 'star', 'theory', 'telescope', 'p
hysics', 'quantum', 'black', 'hole']
8 :  ['cell', 'dna', 'gene', 'genome', 'virus', 'stem', 'genetic', 'bacte
ria', 'molecule', 'tissue']
9 :  ['ocean', 'fish', 'animal', 'sea', 'coral', 'whale', 'marine', 'spec
ie', 'underwater', 'deep']
10 :  ['student', 'teacher', 'education', 'classroom', 'game', 'class',
'learning', 'math', 'grade', 'college']
11 :  ['music', 'song', 'musician', 'instrument', 'musical', 'listen', 's
ing', 'piano', 'concert', 'video']
12 :  ['car', 'vehicle', 'road', 'mile', 'driver', 'traffic', 'street',
'lane', 'transportation', 'highway']
13 :  ['africa', 'african', 'continent', 'hiv', 'aid', 'south', 'nigeri
a', 'kenya', 'leader', 'poverty']
14 :  ['cancer', 'tumor', 'breast', 'drug', 'disease', 'protein', 'bloo
d', 'treatment', 'cell', 'lung']
15 :  ['art', 'artist', 'image', 'painting', 'film', 'paint', 'museum',
'object', 'color', 'draw']
16 :  ['girl', 'mother', 'boy', 'father', 'parent', 'daughter', 'sex', 'b
aby', 'god', 'mom']
17 :  ['building', 'architecture', 'architect', 'material', 'structure',
'site', 'air', 'neighborhood', 'wall', 'apartment']
```

The derived topics from NMF and LDA are displayed above. LDA for this TED Talks dataset produces some of the topics with noisy data and are hard to interpret. I'd say the NMF was able to find more meaningful topics in this dataset. So used the NMF model for final topic modeling and got the top 20 topics along with tags. The final output is

In [54]:
```
ted_topic = pd.DataFrame()
ted_topic['topic'] = [x+1 for x in topic_word.keys()]
ted_topic['topic_tag'] = ['-'.join(x) for x in topic_word.values()]
ted_topic.head()
```

Out[54]:

	topic #	topic tag
0	1	war-government-political-democracy-violence-co...
1	2	patient-health-doctor-disease-medical-hospital...
2	3	planet-earth-mars-solar-energy-atmosphere-star...
3	4	business-economy-economic-market-growth-china-...
4	5	datum-computer-machine-internet-phone-web-digi...

8. Prediction Modeling:

In addition to the Recommendation system, Summarization and Topic modeling, I tried prediction of TED Talk ratings using different machine learning algorithms.

Since target is a categorical variable with many classes and data is labelled data, it is a Supervised multiclass classification problem. The dataset is imbalanced because the classes in target has an unequal distribution. For modeling I choose to work with a machine learning library - scikit.learn.

Metrics: Choosing the right metrics is the key to assess the performance of a model. I choose to take a “weighted” F1 score. For multi-class problems with imbalance data, we have to average the F1 scores for each class. The weighted F1 score averages the F1 score for each class by taking the class imbalances into account. In other words, the number of occurrences of each class does figure into calculation when using “weighted” score.

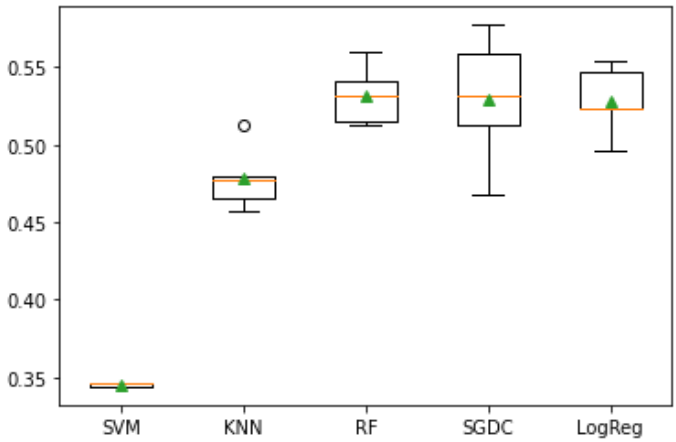
First, I encoded all the categorical variables using Multilabelbinarizer and Onehotencoder and split the data into train, test sets. And created a word vectorizer using Tfidf Vectorizer on transcript. The challenge is that the matrix gendered from Tfidf is very sparse (or high dimension) and noisy (or includes lots of low frequency words). So truncated SVD is adopted to reduce dimension.
The idea of SVD is finding the most valuable information and using lower dimensions to represent the same thing. I built a LSA model for dimensionality reduction and got the below results

```
TF-IDF train output shape: (1718, 37716)
TF-IDF test output shape: (737, 37716)
LSA train output shape: (1718, 50)
LSA test output shape: (737, 50)
Trainging data shape: (1718, 4079)
Test data shape: (737, 4079)
Sum of explained variance ratio: 14%
```

It would be a good idea to spot check a suite of different nonlinear algorithms on a dataset to quickly flush out what works well and deserves further attention and what doesn't. I evaluated the following machine learning models:

Support Vector Machine (SVM)
k-Nearest Neighbors (KNN)
LogisticRegression (LogReg)
Random Forest (RF)
SGDClassifier (SGD)

The output results are:



The two models Random Forest classifier and SGDClassifier gave the almost similar results. So, I tried both algorithms for the best results. Because our dataset is severely imbalanced, we are going to try other imbalanced methods. There are four types of imbalanced classification techniques to spot check:

- Cost-Sensitive Algorithms
- Data Resampling Algorithms
- One-Class Algorithms
- Probability Tuning Algorithms

Here I tried only the Cost-Sensitive Algorithm only.

Cost-Sensitive Algorithms Cost-sensitive algorithms are modified versions of machine learning algorithms designed to take the differing costs of misclassification into account when fitting the model on the training dataset.

Choosing the best model:

Out[72]:

	Model	f1_score(weighted)	Precision	Recall	Training Accuracy	Testing Acuuracy
0	Random Forest	0.48	0.46	0.54	1.00	0.54
1	CostSenstivity with RF	0.49	0.51	0.54	1.00	0.54
2	SGDClassifier	0.53	0.55	0.56	0.57	0.56
3	CostSenstivity with SGD	0.42	0.54	0.42	0.49	0.41

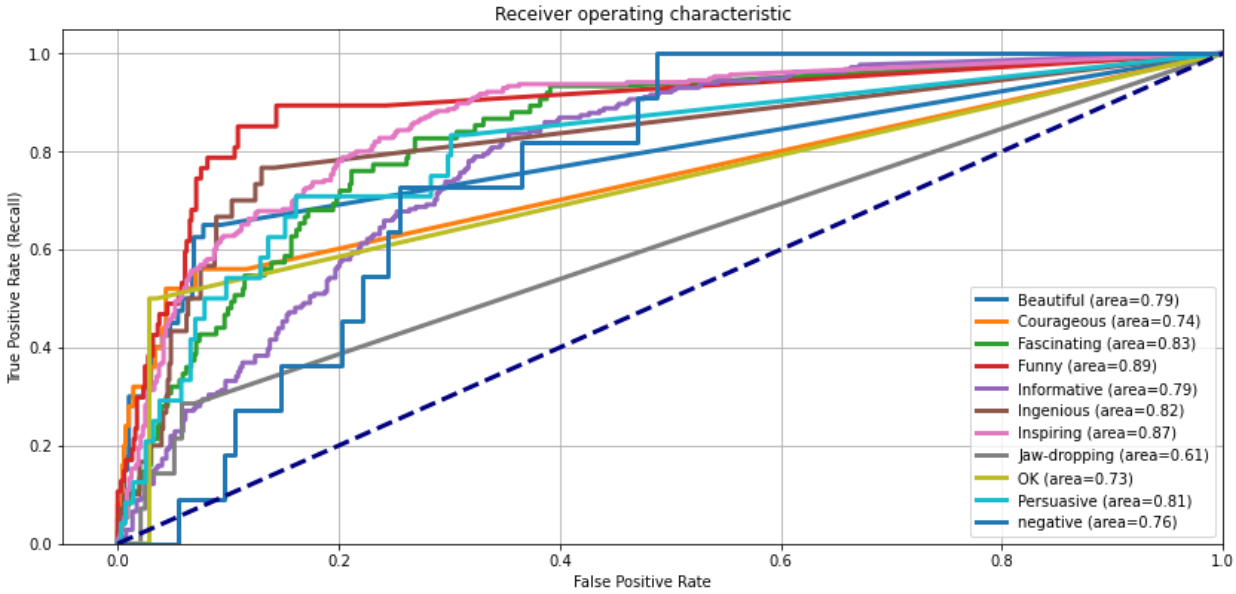
From the above results I choose Cost sensitivity with random forest algorithm gave better accuracy of 49% compared to other models. So, I choose the Cost sensitivity with random forest algorithm as the final model. After that I did hyperparameter tuning to increase the performance of the model and the model performance was increased by 1% and gave the final f1_score(weighted) as 50%. The best model metrics are

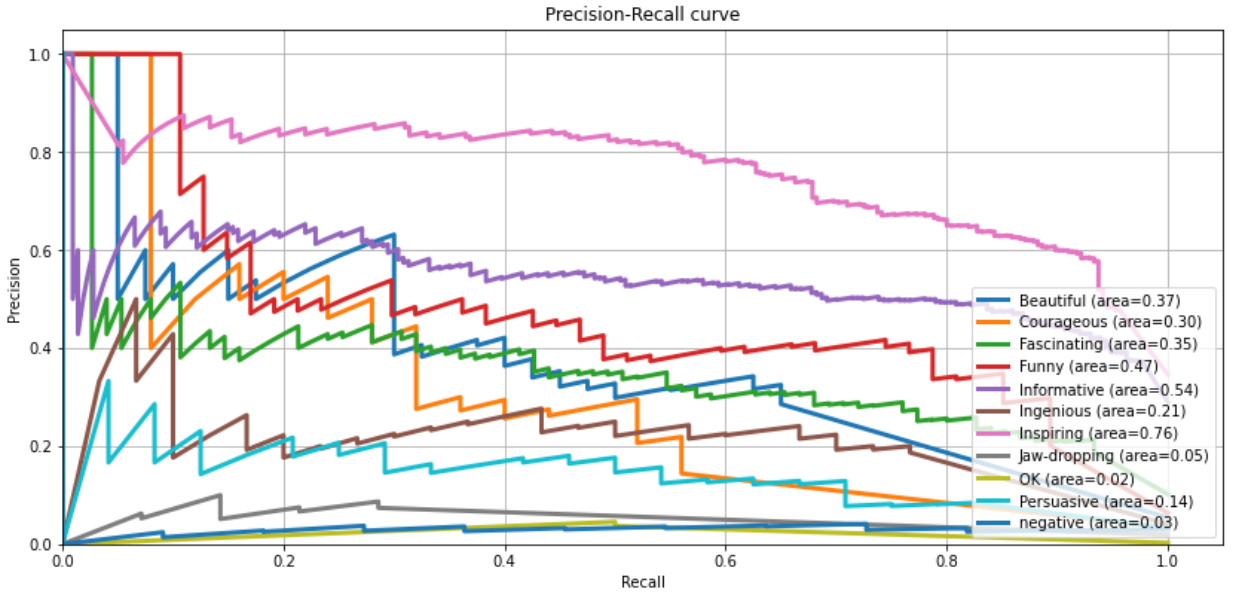
Out[73]:

	Model	Training_accuracy	Test_accuracy	f1_score(weighted)	Precision	recall	Hyperparan
0	CostSenstivity with RF	1.0	0.53	0.5	0.49	0.53	n_estim min_sample : 4,

AUC-ROC and Precision-Recall for Multi-Class Classification

The AUC-ROC curve is only for binary classification problems. But we can extend it to multiclass classification problems by using the One vs All technique.

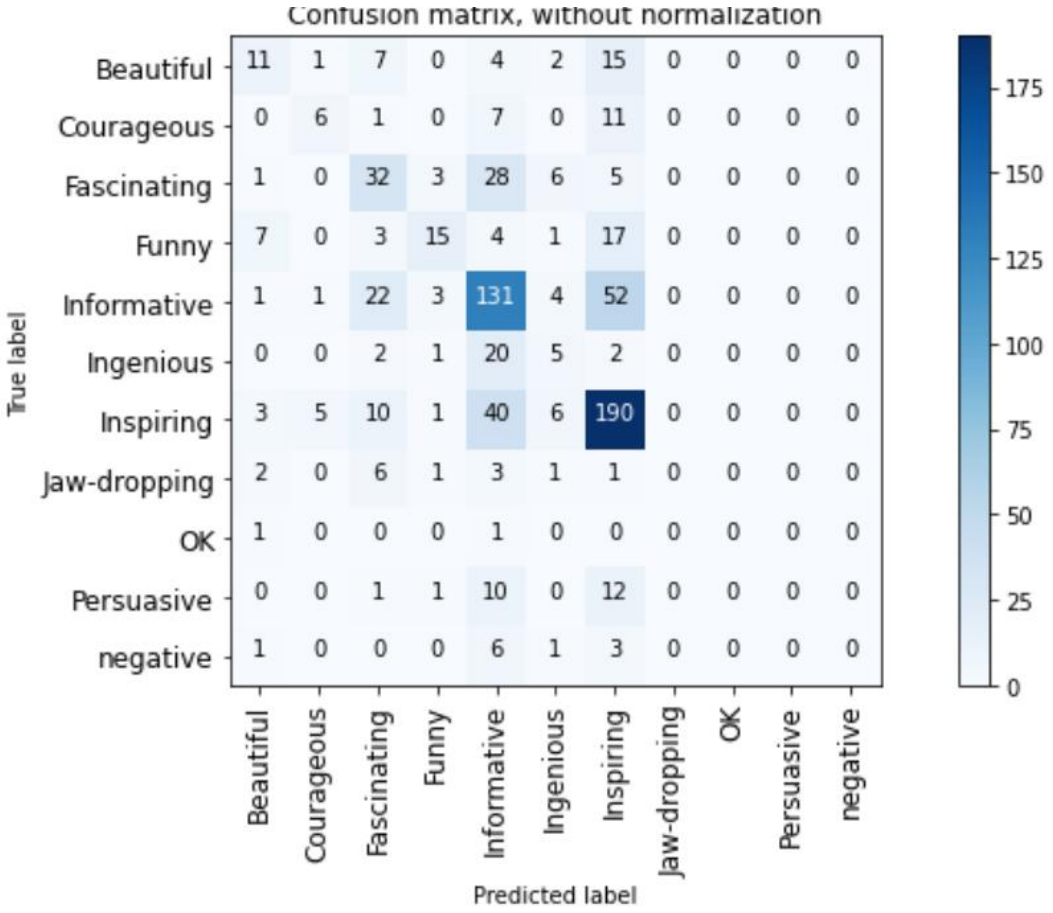




Predictions

Next, I reviewed the predictions using a confusion matrix. A confusion matrix is a table that is often used to describe the performance of a classification model (or "classifier") on a set of test data for which the true values are known.

Confusion matrix



9. Conclusion:

Our work started with merging the two datasets ted_main.csv and transcripts.csv and then the data cleaning process, during which we changed the original order of data columns for convenience and date columns from the Unix timestamps into a human readable format timestamp. And then checked for the Null values, duplicates and dropped the duplicated rows. In the Exploratory Data Analysis section, we analyzed the dataset using plots such as bar plots, box plots and histograms. Furthermore, this section has figured out other significant analysis about our dataset, regarding the most viewed talks of all time, the top 10 speakers and speaker's occupations. We also made hypotheses to figure out the relation

between views and speakers' occupation. From the ANOVA test, we concluded that **There is no statistical significant relationship between views and speakers occupation**. We also showed interesting statistics about views, comments distribution and proved their relationship using Pearson correlation statistical test. And we also showed TED Talks distribution over years, months and weekdays, and some of them were a bit surprising. During the analysis we figured out the outliers and did not remove them as they are actual data for our exploratory analysis. We figured out the collinear features through a heat map. We also analyzed several other pairs for a meaningful correlation, but they do not seem to be strongly correlated. We showed the duration distribution and observed that the short duration TED talks are more famous, it is more likely that people are interested in shorter duration talks because they are able to grasp the talk's content easily or they don't have time to watch longer duration talks. We also analyzed the ratings features and visualized the top 10 most funniest, beautiful, inspiring, jaw-dropping and confusing talks of all time. We investigated the TED word cloud to know about which words are most often used by TED Speakers as well as TED themes and occupations.

Next, we moved on to the preprocessing step, there we did feature extraction and feature engineering on the dataset. And then we did text preprocessing on the transcript which includes converting all letters to lower or upper case, converting numbers into words or removing numbers, removing punctuations, accent marks and other diacritics, removing white spaces, removing stop words, sparse terms and particular words. We did sentiment analysis on transcript and derived appropriate rating categories for transcripts from rating feature. And then we visualized the ratings categories with respect to sentiment.

Next we applied summarization algorithms using spaCy, Gensim and sumy (LexRank, LSA, etc) to extract the summary from the transcript. We found that summarization with spaCy gave good results compared to others for this dataset. And then we moved onto the Recommendation system, where we created word vectorizer using Tfidf Vectorizer on transcript, then calculated the cosine similarity to find out how similar the TED Talks are to each other and then we built a recommender function based on cosine similarity to get the top 10 most similar talks.

Next, we applied Topic Modelling algorithms LDA (Latent Dirichlet Allocation) and NMF (Non-negative Matrix Factorization) on the transcript data to derive the topics around which most of the famous TED talks are presented. We identified NMF gave better results than LDA for this dataset. Finally, we moved on to the prediction model to predict the rating of TED Talks. On applying predictive modelling on the final dataset, we find that of all classifiers used to train the model, Random Forest gives the best results with an accuracy score of 54% and F1(weighted) score of 50%. We used various classifiers like Random Forest, SVM, Logistic Regression and SGDClassifier on 80:20 split to train and test the model and Random Forest outperformed among all.

In conclusion, our work led to interesting results, analysis and statistics, but also provided useful tools both for audience and speakers, which allows a better understanding of TED Talks dataset.

This project gave me an opportunity to explore this freely available dataset using NLP and a proper data science pipeline of data wrangling, data analysis, data visualization, prediction, and data storytelling.

10. Future Improvements

- The recommendation engines used by the official ted page, will be a degree of magnitude more sophisticated than what we demonstrated here and would also involve use of some sort of historical user-item interaction data. Would love to try the TED Talk recommendation system using historical user-item interaction data if available.
- Further analysis can be done over the rating column in the dataset to relate the negative comments with topics of TED talk and find the area of talk which has received more negative feedback.
- We can also make some more analysis over topic and area of TED Talk, by combining some other datasets like news article, social media post etc. to find for any pattern between how the hot discussed topics over world found from news article and social media post are included in TED talk topics, around the same time frame as of the hot discussion over the world.
- Further we can use the most advanced technologies like deep learning and Neural Networks to boost the accuracy of our prediction model.