Udacity Machine Learning Engineer Nanodegree: Capstone Report

Holf Yuen

December 22nd, 2017

What Makes a Popular TED Talk?

I. Definition

Project Overview

Founded in 1984, TED (Technology, Entertainment and Design) is a nonprofit devoted to spreading ideas, usually in the form of short, powerful talks (18 minutes or less). TED conference began in 1984 and nowadays its topics extend from origin "TED" to almost all topics (from science to business to global issues). World renowned figures such as Al Gore, Bill Gates and Elon Musk have been invited as speakers and the talks are available in more than 100 languages. Since the recordings of the talks are available online in ted.com, TED talks gain massive popularity.

Being well-known by its slogan "Ideas worth spreading" and the time limit of 18 minutes, TED talks become an important source of inspiring ideas, stories and discoveries. It is more and more common that a speaker, author or thinker be introduced by his or her TED talk.

There was previous machine learning analysis effort made by Pang and Vadivelu on Kaggle. Statistical analysis of past TED talks also appeared in the paper of Sugimoto et al (2013).

The dataset to use is "TED Talks" dataset on Kaggle, with data on 2550 TED talks (including independently run TEDx events) uploaded to TED.com until September 21st, 2017. The dataset consists of two files. The first file, **ted_main.csv**, contains descriptive and numerical data of each talk. The second file, **transcripts.csv**, contains transcripts of each of the talks. In this project, only ted_main.csv is used.

The prediction model is built based on features including **title, description, occupation of speaker, number of speakers, duration of talk, event, date of filming and publishing, number of comments, tags (which indicate topics), number of languages, and views count**.

Problem Statement

The project intends to predict whether a TED talk will be popular or not. This is a **classification** problem.

TED users can give ratings to each talk. There are 14 possible ratings and they will be categorized as positive, negative and neutral:

- Positive: 'Beautiful', 'Courageous', 'Fascinating', 'Funny', 'Informative', 'Ingenious', 'Inspiring', 'Jaw-dropping', 'Persuasive'
- Negative: 'Confusing', 'Longwinded', 'Obnoxious', 'Unconvincing'
- Neutral: 'OK'

In this project, a "popular" TED talk is defined by its ratio of positive to negative ratings. Transformation is made to avoid "divided by zero" error. If the ratio is above 5 it is defined as "Popular"; otherwise it is "Not Popular".

To build the prediction model, machine learning algorithms including logistic regression, decision trees, support vector machines, Gaussian Naive Bayes, neural networks, and ensemble learning (such as Adaboost and random forests) will be explored.

Natural language processing techniques are used to extract features of text columns to provide inputs for the model.

The benchmark model will be the logistic regression model using inputs other than text data (i.e. title, speaker occupation, description, and tags). Text data requires more processing than numerical and categorical data and so the project intends to explore whether a model with text data will beat a simple model without that.

Metrics

F-1 score is used as evaluation metric. F-1 score is calculated as:

F1 = 2 * (precision * recall) / (precision + recall)

Where

Precision = True Positive / (True Positive + False Positive)

And

Recall = True Positive / (True Positive + False Negative)

As it is a classification model with unbalanced class (only 301 out of 2439 talks will be classified as "Not Popular"), F-1 score is an appropriate measure. This score is not biased in favor of precision and recall, which is suitable in this project.

II. Analysis

Data Exploration

In the dataset, there are 2550 rows and 17 columns. The first few lines look like this:

```
   comments                                        description  duration  \
0      4553  Sir Ken Robinson makes an entertaining and pro...      1164
1       265  With the same humor and humanity he exuded in ...       977
2       124  New York Times columnist David Pogue takes aim...      1286
3       200  In an emotionally charged talk, MacArthur-winn...      1116
4       593  You've never seen data presented like this. Wi...      1190
```

```
     event   film_date  languages   main_speaker  \
0  TED2006  1140825600         60   Ken Robinson
1  TED2006  1140825600         43        Al Gore
2  TED2006  1140739200         26    David Pogue
3  TED2006  1140912000         35  Majora Carter
4  TED2006  1140566400         48   Hans Rosling

                                        name  num_speaker  published_date  \
0      Ken Robinson: Do schools kill creativity?            1      1151367060
1           Al Gore: Averting the climate crisis            1      1151367060
2                  David Pogue: Simplicity sells            1      1151367060
3             Majora Carter: Greening the ghetto            1      1151367060
4  Hans Rosling: The best stats you've ever seen            1      1151440680

                                             ratings  \
0  [{'id': 7, 'name': 'Funny', 'count': 19645}, {...
1  [{'id': 7, 'name': 'Funny', 'count': 544}, {'i...
2  [{'id': 7, 'name': 'Funny', 'count': 964}, {'i...
3  [{'id': 3, 'name': 'Courageous', 'count': 760}...
4  [{'id': 9, 'name': 'Ingenious', 'count': 3202}...

                                       related_talks  \
0  [{'id': 865, 'hero': 'https://pe.tedcdn.com/im...
1  [{'id': 243, 'hero': 'https://pe.tedcdn.com/im...
2  [{'id': 1725, 'hero': 'https://pe.tedcdn.com/i...
3  [{'id': 1041, 'hero': 'https://pe.tedcdn.com/i...
4  [{'id': 2056, 'hero': 'https://pe.tedcdn.com/i...

                     speaker_occupation  \
0                       Author/educator
1                      Climate advocate
2                  Technology columnist
3    Activist for environmental justice
4  Global health expert; data visionary

                                                tags  \
0  ['children', 'creativity', 'culture', 'dance',...
1  ['alternative energy', 'cars', 'climate change...
2  ['computers', 'entertainment', 'interface desi...
3  ['MacArthur grant', 'activism', 'business', 'c...
4  ['Africa', 'Asia', 'Google', 'demo', 'economic...

                         title  \
0      Do schools kill creativity?
1      Averting the climate crisis
2                 Simplicity sells
3              Greening the ghetto
4  The best stats you've ever seen

                                                 url     views
0  https://www.ted.com/talks/ken_robinson_says_sc...  47227110
1  https://www.ted.com/talks/al_gore_on_averting_...   3200520
2  https://www.ted.com/talks/david_pogue_says_sim...   1636292
```

```
3  https://www.ted.com/talks/majora_carter_s_tale...   1697550
4  https://www.ted.com/talks/hans_rosling_shows_t...  12005869
```

Note that 'related talks' and 'url' are not used in this project and are dropped early on. The 'name' column is merely combining 'main speaker' and 'title', so it is also dropped. The 'main speaker' column is not used for building prediction model as well.

The 'ratings' column contains is a JSON object containing the count of all 14 ratings. It needs to be processed into a table form of numerical variables before calculating the ratio of positive to negative ratings. **Talks with the ratio above 5 are classified as "Popular"**.

```
   Beautiful  Confusing  Courageous  Fascinating  Funny  Informative  \
0       4573        242        3253        10581  19645         7346
1         58         62         139          132    544          443
2         60         27          45          166    964          395
3        291         32         760          132     59          380
4        942         72         318         4606   1390         5433

   Ingenious  Inspiring  Jaw-dropping  Longwinded    OK  Obnoxious  \
0       6073      24924          4439         387  1174        209
1         56        413           116         113   203        131
2        183        230            54          78   146        142
3        105       1070           230          53    85         35
4       3202       2893          3736         110   248         61

   Persuasive  Unconvincing
0       10704           300
1         268           258
2         230           104
3         460            36
4        2542            67
```

The following table summarizes the format of other feature variables:

Feature	Description	Variable Type
title	The title of the talk	text
description	A blurb of what the talk is about	text
speaker_occupation	The occupation of the main speaker	text
num_speaker	The number of speakers in the talk	numerical
duration	The duration of the talk in seconds	numerical
event	The TED/TEDx event where the talk took place	text
film_date	The Unix timestamp of the filming	datetime
published_date	The Unix timestamp for the publication of the talk on TED.com	datetime
comments	The number of first level comments made on the talk	numerical
tags	The themes associated with the talk	text
languages	The number of languages in which the talk is	numerical

	available	
views	The number of views on the talk	numerical

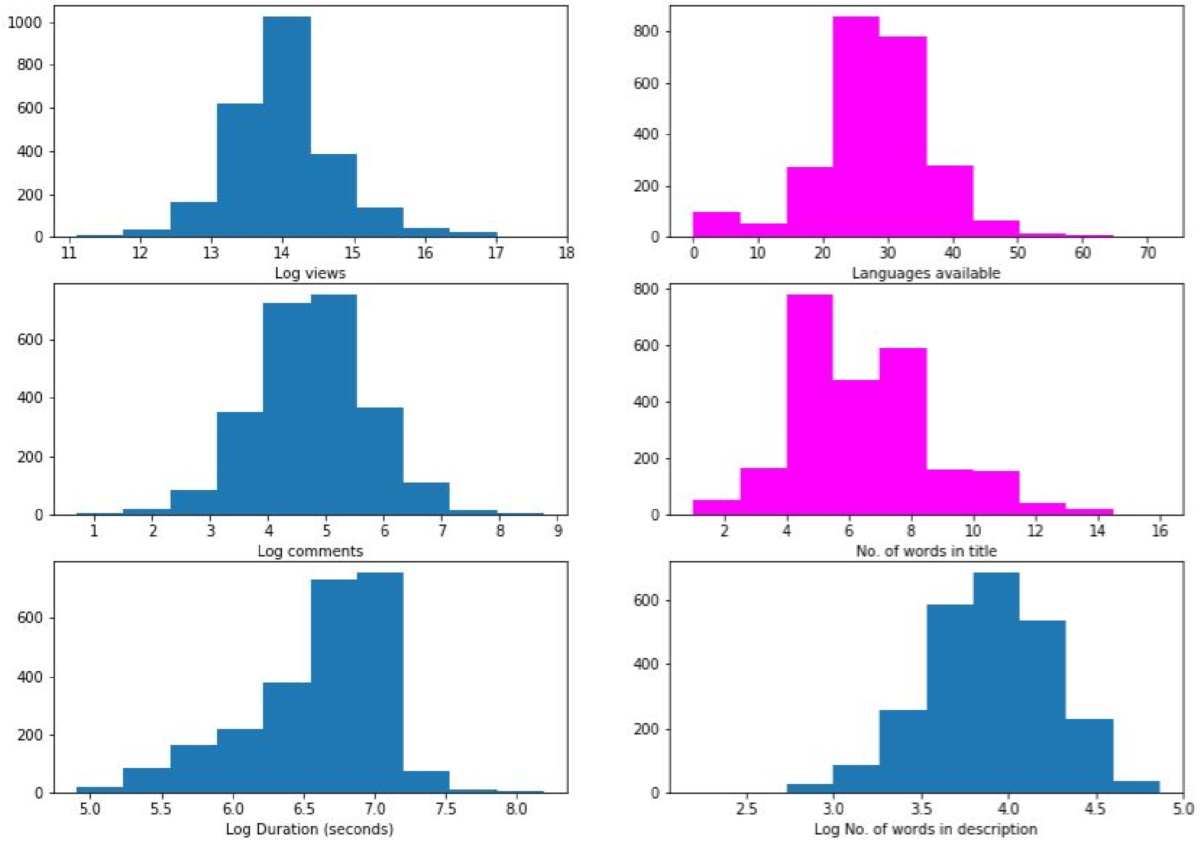
Among 2550 talks in the dataset, some are in fact not TED or TEDx events (for example, there is a video filmed in 1972, even before TED is established). There were 111 of them and they are excluded. 2439 talks remain in the dataset. Among them, 2138 talks or 87.66% are classified as "Popular" (with ratio of positive to negative rating above 5).

Below is the summary of other inputs features:

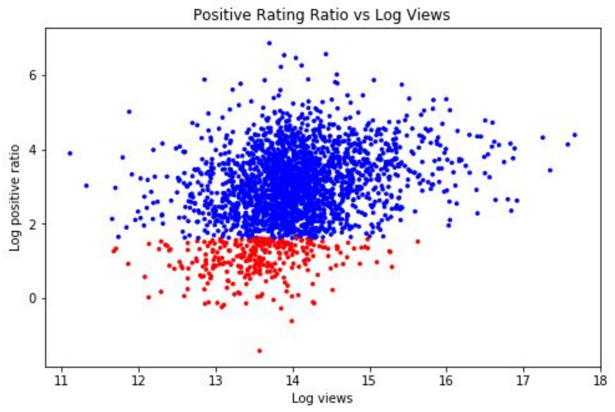
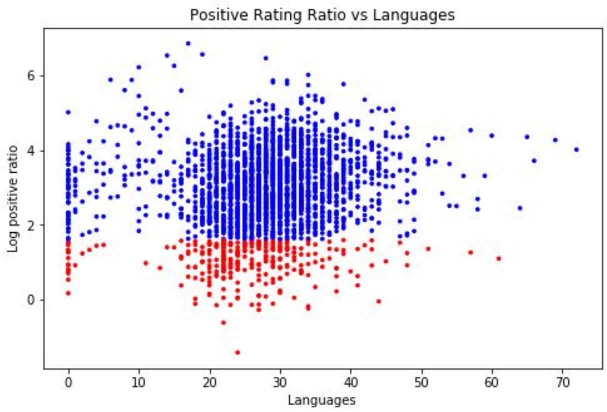
- The talks have a mean of 192 comments. Standard deviation 285.67;
- Length of talks range from 135 seconds (2 mins 15 seconds) to 3608 seconds (1 hour and 8 seconds). Mean is 811 seconds and standard deviation 331.88 seconds. 482 talks exceeded the "classic" time limit of 18 minutes (1080 seconds);
- 58 talks have more than one speakers; 86.21% of talks by single speaker are “Popular”, the ratio is 87.69% for talks spoken by more than one speakers;
- The talks are available in an average of 27.7 languages. The highest number is 72;
- The talks have an average view count of 1.73 million, with the highest viewed 47.23 million times;
- Filmed dates and published dates are presented in Unix timestamp format. They are converted to date objects. The earliest talk is filmed in February 2nd, 1984 and the latest August 27, 2017. The published dates range from June 27, 2006 to September 22, 2017;
- 1968 talks are TED events and 471 from TEDx events; Among TED events, 87.20% is classified as “Popular”; among TEDx events, 89.69% is classified as “Popular”.

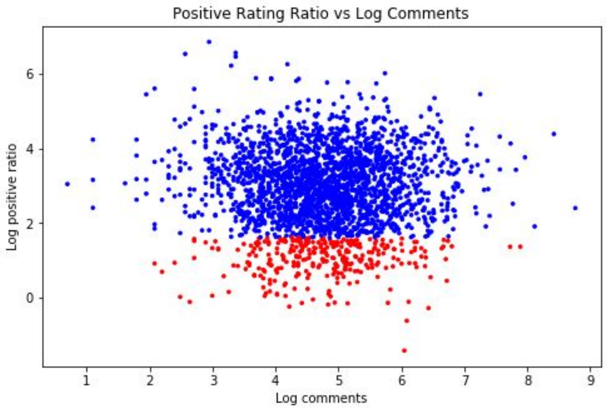
Exploratory Visualization

Below is the histograms of various numerical variables. The “languages available” and “number of words in title” without log transformation look more like a normal distribution. For “views”, “comments”, “duration” and “number of words in description”, as they have some extreme values, a logarithmic transformation of those variables show a more normal-shaped distribution.

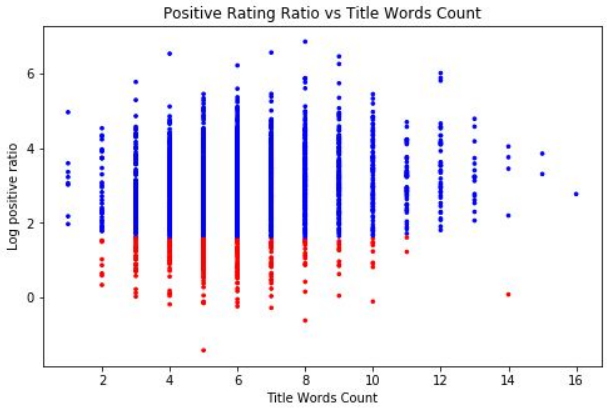


The following plots are ratio of positive to negative ratings against various numerical variables. Blue dots are classified as “Popular” and red dots “Not Popular”:

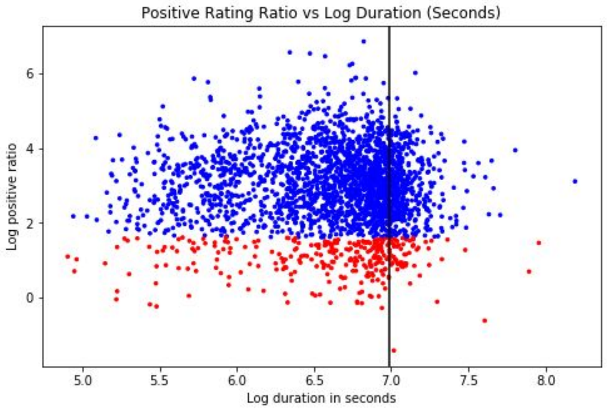
	
At around 16 or above in x-axis (exp(16) ~ 6 million), all talks are popular	No discernable pattern on popularity



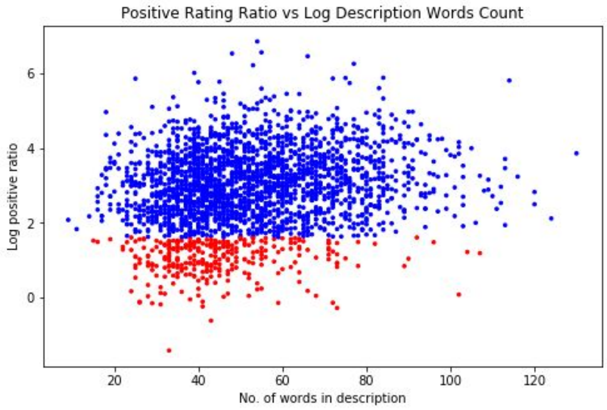
No discernable pattern on popularity



All talks with single-word titles and all except 1 for talks with titles of 12 words or more are "Popular". On closer inspection, single-worded talks are musical performances.



The vertical line indicates classic "18-minute" time limit of TED talks. No discernable pattern on popularity



No discernable pattern on popularity

The following tables show the ratio of "Popular" talks by months and days of week. It is worth noting that talks filmed in January and May are 96.6% and 93.5% popular respectively, outperforming the average of 87.66% by a wider margin. Talks filmed on Saturday have a lower popular rate of 81.4%.

	Filmed	**Published**
Jan	0.965517	0.870000
Feb	0.821369	0.881443
Mar	0.896203	0.900452
Apr	0.894737	0.845528
May	0.935185	0.882353
Jun	0.894737	0.873874
Jul	0.858300	0.868687

Aug	0.827586	0.900585
Sep	0.910891	0.874439
Oct	0.909574	0.853774
Nov	0.891304	0.873563
Dec	0.894737	0.910828

	Filmed	**Published**
Monday	0.905724	0.829609
Tuesday	0.895570	0.884615
Wednesday	0.893910	0.890351
Thursday	0.869811	0.875000
Friday	0.878788	0.885650
Saturday	0.813880	0.883721
Sunday	0.869159	0.920000

Algorithms and Techniques

The following supervised learning algorithms are explored in this report. They are all suitable for classification problem in this project.

- Logistic regression: It is a linear model to fit the regression line using log-odds of one class as dependent variable against features variables.
- Decision trees: It is a non-parametric model for classification to predict a class based on decision rules learnt from data
- Support Vector Machines: It is to fit a linear or nonlinear boundary in multidimensional space to separate on class from another
- Gaussian Naive Bayes: It applies Bayes theorem with “naive” assumption of independence between each pair of features. It fits a model which has the maximum likelihood of seeing our data with existing labels. Gaussian Naive Bayes assumes the likelihood of the features are Gaussian
- Neural Network - Multilayer perceptrons: It learns a function to transform the inputs into a layer of intermediate output (or hidden layer of neutrons), which in turn are used to predict the class of output
- Random forests: It builds an ensemble of decision tree classifiers and predict by averaging the prediction of diverse prediction trees.
- Adaboost: It fits a sequence of weak learners (such as small decision trees) on repeatedly modified versions of data. On each successive iteration, the sample weights are individually modified such that the weights of previously misclassified are increased.

The dataset from Kaggle is quite clean and tidy. Therefore not much effort is needed in coding to clean the irregularities of data. Some complicated fields such as “ratings” and “tags” can be readily processed as a JSON object and Python list respectively. However, a little complication exists in below fields:

- Date: The “film_date” and published_date” are recorded as Unix timestamps in original dataset. They need to be converted to normal datetime object for further analysis;
- Occupation: The field is a free form text and some irregularities need to be cleared. For example, there are some unnecessary spaces to be removed, and the texts need to be converted to lower case. For speakers with multiple occupations, their occupations can be separated by comma (,), semicolon (;) or “and” word. To separate the different occupations and convert into tags, some regular expression is used in the coding process.

Benchmark Model

The benchmark model chosen is a logistic regression model using inputs other than text data (including title, speaker occupation, description, tags and transcripts). Default parameters in scikit-learn library are used.

The benchmark model has an accuracy of 87.05% with F-1 score 0.9308 on the testing set. As a side note, a naive predictor classifying all talks as “Popular” has a accuracy of 87.38% and 0.9326 F-1 score.

III. Methodology

Data Preprocessing

For structured (non-text) inputs, the following preprocessing transformation are made:

Feature	Transformation
num_speaker	Convert to “single_speaker” variable with value 1 if only 1 speaker in the talk, and 0 if multiple speakers
duration	Apply log transformation, then min-max scaling (with minimum -1 and maximum 1)
event	Convert to categorical data of “TED” and “TEDx”
film_date	Extract categorical variables ‘month’ and ‘day of week’
published_date	Extract categorical variables ‘month’ and ‘day of week’
comments	Apply log transformation, then min-max scaling (with minimum -1 and maximum 1)
languages	Min-max scaling (with minimum -1 and maximum 1)
views	Apply log transformation, then min-max scaling (with minimum -1 and maximum 1)

Text columns, including tags, speaker occupation, title and description, are preprocessed to extract features as model inputs:

Tags:

Each row is a list of tags. For example: the first line is:

> ['children', 'creativity', 'culture', 'dance', 'education', 'parenting', 'teaching']

I loop through the column to find the most frequent tags. The top 3 tags are “technology”, “science” and “global issues”, which occur 701, 534 and 478 times respectively. All tags with 20 or more occurrences are used as feature variables. For each tag, the value is 1 if the tag exists in the talk, and 0 otherwise. There are 209 tags with 20 or more occurrences.

Speaker occupation:

There column is a free form text variable, and a speaker can report more than one occupations, so preprocessing is necessary. The text for each row is split into a list of “occupation tags” to try to capture each occupation. Some special treatments are made as below:

- There are “singer/songwriter”, “singer-songwriter” and “singer, songwriter” reported for different speakers. They are all assigned “singer” and “songwriter” tags
- Some speakers report “activist” as their occupation and some are more specific (such as “education activist” or “environmental activist”). They are assigned “activist” tags. Similar treatment is made for “writer”, “author” and “artist”

The 3 most frequent occupation tags are 'activist' (139 occurrences), “artist” (102) and “writer” (99). All occupation tags with 20 or more occurrences are used as feature variables. For each occupation tag, the value is 1 if the tag exists in the talk, and 0 otherwise. There are 24 tags with 20 or more occurrences.

All talks with tags “senses”, “physiology”, “empathy”, “water”, “depression”, “gender” and “success” and occupation “neuroscientist” are “Popular” in our dataset. On the other hand, talks with tags with “cars” have almost half (45%) talks classified as not “Popular”, and talks with occupations “economist”, “philosopher” and “designer”, and tags “industrial design”, “security” and “religion” have proportion of “Popular” label below 75%.

Title:

Three features are extracted from the 'title' field:

- Word count (min-max scaling applied with minimum -1 and maximum 1)
- Square of word count (min-max scaling applied with minimum -1 and maximum 1) - this is used to try to capture the relationship that talks with 1-word title and titles or 12 or more words are almost all “Popular”.
- Whether the title is a question. The 'title' is tokenized. If there is a question mark (?) or one of the following words: 'what', 'how', 'when', 'why', 'which' and 'where', it is regarded as a question

Description:

Word count is extracted as a feature variable. It is than transformed by logarithmic transformation and min-max scaling (minimum -1 and maximum 1)

Implementation

Python version 2.7 is used for coding the machine learning algorithm and the following Python packages are used:

- numpy
- pandas
- matplotlib
- datetime
- time
- re
- nltk
- sklearn

The machine learning algorithms are run on a Chromebook with 4GB of RAM.

Categorical variables (month and day-of-week the talk is filmed and published, and event category), are one-hot encoded into columns of 1s and 0s where 1 is true and 0 is false.

Redundant columns of categorical variables (including one each from “filmed month”, “filmed day of week”, “published month”, “published day of week”, and “event category”) are removed to avoid multicollinearity problem.

I finally decide not to include “views” and “comments” as features into the prediction model. See Reflection section on more details.

The input data is split into training and testing set with 75:25 ratio. The training set has 1829 talks and testing set 610. Then each algorithm is in turn fit against training data and tested with testing data.

Refinement

Among training data, 5-fold grid-search cross-validation is used to pick the best hyperparameters for each algorithm. The following table lists the hyperparameters tested for grid-search cross-validation for each algorithm:

Algorithm	Hyperparameters tested in cross-validation	Final hyperparameters picked for optimization
Benchmark model (Logistic regression on non-text data)	Not applicable	Not applicable
Logistic regression	'C': [0.01, 0.05, 0.1, 1] 'solver': ['newton-cg', 'lbfgs', 'liblinear', 'sag']	‘C’: 0.01 ‘solver’: ‘newton-cg’
Decision trees	'criterion': ['gini', 'entropy'] 'max_features': [None, 'auto', 'log2'] 'min_samples_split': [2,3,5]	'criterion': 'entropy' 'max_features': None 'min_samples_split': 3
Support vector machine	'C': [0.05, 0.1, 0.5, 1] 'kernel': ['linear', 'poly', 'rbf', 'sigmoid'] 'degree': [2, 3]	'C': 0.05 'kernel': 'linear' 'degree': 2
Random forest	'n_estimators': [5, 10, 15]	'n_estimators': 15

	'criterion': ['gini', 'entropy'], 'max_features': [None, 'auto', 'log2'] 'min_samples_split': [2,3,5]	'criterion': 'gini' 'max_features': 'auto', 'min_samples_split': 5
Gaussian Naive Bayes	Not applicable	Not applicable
Neural Network - Multilayer perceptrons	'hidden_layer_sizes': [(4,), (8,), (16,), (32,)] 'activation': ['logistic', 'tanh', 'relu'] ‘learning_rate_init': [0.001, 0.01, 0.1, 1]	'hidden_layer_sizes': (32,) 'activation': 'relu' ‘learning_rate_init': 0.1
Adaboost	'n_estimators': [10, 20, 50, 100] 'learning_rate': [0.001, 0.01, 0.1, 1]	‘n_estimators’: 10 ‘learning_rate’: 0.01

IV. Results

Model Evaluation and Validation

The testing set accuracy and F-score of each supervised learning algorithm are listed in the following table. The metrics of benchmark model and “naive” predictor of all “Popular” are also tabulated below for comparison:

Prediction	Accuracy	F-1 score
All “Popular”	0.8738	0.9326
Benchmark model (Logistic regression on non-text data)	0.8705	0.9308

Algorithm	Accuracy (with cross-validation)	F-1 score (with cross-validation)	Total run time (seconds, including grid-search CV)
Gaussian Naive Bayes	0.3164	0.3748	N/A
Logistic Regression	0.8738	0.9326	21.5487
Decision Trees	0.7934	0.8816	7.1985
Support Vector Machine	0.8738	0.9326	265.9068
Random forest	0.8754	0.9335	120.4178
Neural Network - Multilayer perceptrons	0.8738	0.9326	37.3952
Adaboost	0.8738	0.9326	65.7838

Note: (1) Cross-validation is not done on Gaussian Naive Bayes as there is no hyperparameter to tune (2) All random seeds are set to 108 where applicable

It turns out that only random forest model can achieve marginally better score than both the benchmark model and the “naive” prediction of claiming all talks “Popular”. On closer inspection the random forest model correctly predicts only 1 out of 77 talk that is not “Popular” in the testing set, while classifying all “Popular” talks correctly.

Note that logistic regression, support vector machine, neural network and Adaboost all return a prediction model saying all talks are “Popular”. This is unsatisfactory and may indicate the current data is not enough to build a prediction model to classify talks as “Popular” or not.

Given the dataset size of 2439 talks covering a wide range of dates, topics, and events, and no significant outliers related to feature variables, I believe the model can generalize well to unseen data and robust to small changes in data. On a re-run of the project using another random seed of 1222 in train_test_split and 1234 in training the machine learning models, it comes up the same result that only random forest can beat the benchmark model by a small margin (F-1 score of 0.9342 vs 0.9335 in benchmark model), and logistic regression, support vector machine, neural network and Adaboost still return the “naive” prediction saying all talks are “Popular”.

Justification

Among tested algorithms, only random forest yield slightly higher accuracy and F-1 score than the benchmark model. It indicates that even with features from text data included, **it is difficult to find a model that satisfactorily distinguish “Popular” TED talks from not “Popular” ones**. More data is needed to set up such a prediction model.

For future analyses, random forest is more likely to build a better prediction model than other algorithms. Gaussian Naive Bayes performs poorly in this project and will not be considered in the future.

V. Conclusion

Free-form Visualization

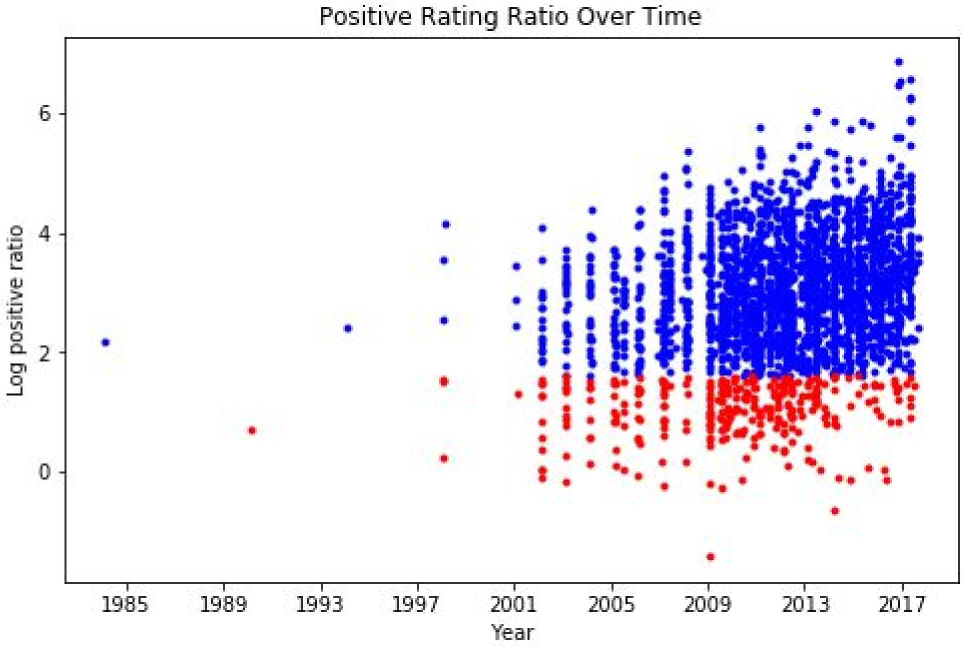
In the random forest model which slightly beats the benchmark, there is only one talk that is correctly predicted as not “Popular”. What is that talk?

Column	**Value**
title	Great cars are great art (5 words)
description	American designer Chris Bangle explains his philosophy that car design is an art form in its own right, with an entertaining -- and ultimately moving -- account of the BMW Group's Deep Blue project, intended to create the SUV of the future. (42 words)
speaker_occupation	Car designer
num_speaker	1
duration	1204
event	TED2002
film_date	2002-02-02 (Friday)
published_date	2007-04-05 (Wednesday)

comments	78
tags	['art', 'business', 'cars', 'design', 'industrial design', 'invention', 'technology', 'transportation']
languages	23
views	867495
pos_ratio	2.34363

We can see that it contains the “most unpopular” tag of “cars” (only 55% with that tag is “Popular”), and “industrial design” is another less popular tag (only 72.4% “Popular”). Its filming month of February and publishing month of April also exhibit a lower than average ratio (82.14% and 84.55%) of talks seen as “Popular”. It is not surprising the talk is predicted as not “Popular”.

Another issue to discuss relates to distribution of positive ratio of talks over time. Below is the chart:



There are a few points far and sparse on the left, which represents talks very early. Some of the most recent talks have exceptionally high log positive ratio, which may indicate they only receive a few ratings. Rating ratios may be more stabilized after more time. This issue is further discussed in “Reflection” section below.

Reflection

I would like to discuss several choices I made throughout the project:

- **Definition of class labels:** I use ratings instead of views as the source of dependent variable for a few reasons. Firstly, ratings more represent preceived quality of the talks. Talks with large view counts can be because they are controversial. Secondly, talks published earlier have more time to grow in views, so it is not fair comparing views of earlier talks and more recent ones. To adjust views by time since published need further judgement and may not be “fairer” either. The same problem exist in ratings therefore a ratio of ratings is used to get around the problem. The reason of using 5 times instead of 1 is firstly a threshold of 1 will make the classes more unbalanced, and secondly, it is based on the judgement that users are relatively not easy to give a bad rating to a talk unless it is really bad. Therefore it takes a higher ratio (5 here) to determine the talk is “Popular”

- **Choice of “month” and “day of week” as feature variables:** “Year” is not useful in the prediction model to predict future talks (for example, we run the model and turns out that talks in 2007 are all “Popular”, but we can never have new talks in 2007 again unless we invent time travelling). “Month” and “day of week” have a moderate number of classes (12 and 7 respectively) and so are used to build prediction models. “Day” has a value of 1 to 31 and there will be too few data in each class in training data, so not as suitable as “month” and “day of week”
- **How to scale and transform numerical variables:** Whether to perform logarithmic transformation depends on how skewed is the data. When there are values that are exceptionally high (which are the cases in comments, views, duration and word counts in description), log transformation will make the data distributed more “normal”. Scaling is necessary to make numerical feature variables comparable with dummy variables which take values 0 and 1. The final model uses min-max scaling with range [-1,1]. Other scaling methods such as min-max scaling with range [0,1] or standardization (convert to mean 0 and standard deviation 1) have been explored but they have no notable impact on results.
- **Whether to “flip” class labels:** In the original project design, “Popular” talks are assigned label 1 and “Not Popular” 0. While it fits intuitively, it has a problem that naively predicts all talks as “Popular” already has a high F-1 score of 0.9326 which is hard to beat. Flipping the labels was tried so that a “naive” model has F-1 score of 0, forcing the algorithm to find other model weights. However it results in poor accuracy and so it is decided not to flip the labels. Considering the fact that other machine learning problems such as spam email detection may face even more imbalanced class labels yet useful models can be built, labeling which class a 1 and which 0 should not be blamed for not getting the right model.
- **How many tags to include:** Tags and occupations with 20 or more occurrences are now included as feature variables. It was considered to include tags seen only 50 times or more. After unsatisfactory modelling results, I include more variables by lower the threshold to 20, but not much improvement is seen.
- **Whether to include earliest and latest talks:** There are 8 talks that are filmed before 2000 and seen as distinct from other talks when plotting against time, and there are some very recent talks which do not receive many ratings (positive or negative). As the old talks are really TED events and the short time online for new talks affect both positive and negative ratings, it is decided those talks are kept in our dataset.
- **Whether to include “views” and “comments” as features:** The reason they are not included in the final model is that, “ratings”, “views” and “comments” (number of comments) evolve over time together after the talks are published. In practice if we want to predict whether a TED talk receives good rating, the prediction should be made at the time the talk is published, when there is no view counts or comments available. Therefore using “views” and “comments” as prediction features is practically inappropriate.

Improvement

If this project is re-run, there can be following ways of improvement in data preprocessing and modelling:

- From “description” column, I only used word count as a feature. There may be more interesting features to be extracted (e.g. by sentiment analysis) that contribute to a popular TED talk;
- The TED talks dataset from Kaggle contains another file which has the transcripts of each talk. Applying natural language processing on the transcripts can extract inputs which may improve the prediction models.
- For definition of “Popular” or not, I aggregated all kinds of rating into one metric of “ratio of number of positive ratings over negative”. Maybe more information and insight can be drawn if we focus on a particular rating label (such as “inspiring” or “courageous”)