Jarrett Perkins
Evan Segaul
David Taylor

Ted Talk Visualization

Process Journal

https://github.com/esegaul/tedtalkvis

Overview and Motivation:

TED Talks are lectures given by researchers, scientists, and cultural commentators about popular ideas in technology, sociology, politics, and everything in between. We are motivated to examine TED Talk data because we are all avid watchers of the videos that they publish of the lectures, and we believe that the topics covered in TED Talks are good indicators of the ideas that are most popular in science and technology at the time of the talks. Our project is designed to better understand TED Talks and the content that they cover. By visualizing various aspects of these talks, we can better understand what society deems important and worth talking about.

Our goals for this project hinge upon labeling TED talks by topic using topic modeling techniques. We want to be able to categorize a wide range of TED talks into different topic areas in order to help with our analysis. From there, we want to see how the prevalence of these topics change over time as well as how these topics are perceived by audiences and by TED themselves.

Related Work:

Much of the work on this project was inspired by our discussion of text data visualization in class. We realized that we could apply topic extraction techniques to the dataset that we had found on Kaggle (described in the “Data” section) and thought that it would be very interesting to visualize ted talks in terms of the topics that we extract from their transcripts. We had some prior knowledge of the mathematical theory behind dimensionality reduction and principal component analysis from other courses, but we had not ever used these techniques in practice. We were inspired by this paper from researchers at Georgia Tech for the application of Non-Negative Matrix Factorization to text data, as it contains some helpful explanations of the underlying theory of NMF and some useful visualizations of the output of the model.

Questions:

With our project, we are mainly trying to understand what topics are popular within TED talks, how has the popularity of each of these topics changed over time, and how are each of these topics perceived by TED viewers. These questions have stayed fairly consistent over the course of our project, though a number of new questions have arisen as we have worked with that data.

In particular, we realized that we had our own set of topics generated by our topic model as well as tags provided by TED. We originally did not want to use these tags, as there was far too much variation within tags for meaningful analysis. However, we realized it would be interesting to visualize how TED labeled these talks with tags and how those breakdown within our topic areas. Thus, an additional question we explored was how TED’s tags were distributed across topics.

Data:

We found the data for our visualization on Kaggle. A Kaggle user had scraped the TED official website to gather data on every talk given up until September 1st, 2017. There are 2 .csv files which we downloaded: one which contains high-level information on each talk such as title, author, and date, and another which contains the transcript of each talk given. The data we downloaded was very clean and organized, and there were very few null values. This meant that we did not have to do much in terms of cleaning up existing data, but we did have to extract the year from each unix time-stamp and the top user “rating” for each talk for ease of use later on with d3. All of our data cleaning and feature engineering was done in a Jupyter Notebook (‘nmf.ipynb’).

Initially, we planned on using just the dataset provided on Kaggle to make our visualizations. However, after learning about topic extraction from text data, we decided to use Non-Negative Matrix Factorization to categorize each TED Talk transcript with one of ten “learned” topics. We attempted using Latent Dirichlet Allocation at first for topic extraction, but found that the transcripts for each talk were too large to get an accurate fit from the model. NMF yielded a much more interpretable topic model after a bit of hyperparameter tuning.To accomplish this, we first applied scikit-learn’s TfidfVectorizer module to each document (talk transcript) to get a sparse matrix containing information on the “importance” of each word in each document. We removed common English stop-words so that the vectorizer would not pick up on words that appear many times in each talk. We then use scikit-learn’s NMF module to fit a Non-Negative Matrix Factorization model to the sparse matrix, requiring that the NMF model distinguish between 10 different talk topics. NMF works by decomposing the TFIDF matrix into the product of two non-negative matrices WH. In this product, the column vectors of W represents the weighting of “importance” each word within a given topic, and the column vectors of H represent the coefficients for each document of each of the ten features (topics). The product WH is a reasonable approximation for our original vectorized sparse matrix. Each resulting feature (topic) is an array of 5 words which were determined by the NMF model to be the 5 most descriptive terms for each given topic. Then, we apply the NMF model transformer to each talk, giving us a predicted topic for each talk. We also included a predicted topic “id” which, for each talk, is an integer in the range [0,9] that represents the talk’s predicted topic. We did this so that we could group by immutable objects (integers) rather than Python lists.

We then join the predicted topic data back onto the high-level data for each talk and output the data to a csv file (‘labelled_data.csv’) for reading into d3. This will allow us to group talks by predicted topics and examine how the other data provided in the dataset changes by topic. For the prototype, we use some “demo” data sets to set up some basic visualizations. In the final project, we will be reading all of the data in at once (from ‘labelled_data.csv’) and using d3 to group and join the data appropriately.

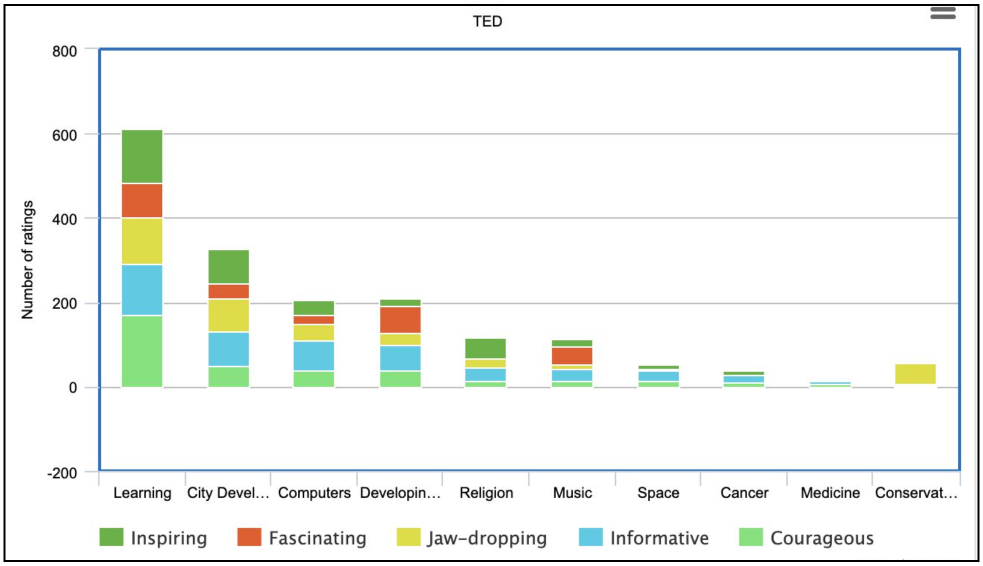
Exploratory Data Analysis:

Our exploratory data analysis was done in a Jupyter Notebook entitled ‘EDA.ipynb’. In the EDA, we started by getting some basic descriptive statistics from the high-level talk dataset, and then plotting some basic relationships between some of the features. We looked at talks published by month, the relationships between comments, views, and number of languages the talk is available in. These were interesting, but ultimately the most impactful part of the EDA was our examination of the “ratings” given to talks by viewers. We had assumed that there would be more diversity in the top rating given to the talks, and that diversity would allow us to create interesting visualizations of talks. However, the work in our EDA shows that the vast majority of talks have a top rating of either “Informative”, “Beautiful”, or “Inspiring”. These terms are vague and don’t reveal very much information about the talks themselves, and it’s obvious that TED Talks are, by nature, supposed to fit into these categories. There are very few talks with a negative top rating (e.g., “Obnoxious” or “Confusing”). This led us to conclude that a graph visualization in which proximity of nodes encodes the similarity of top ratings of two videos would likely be uninformative, as many talks have the same top ratings. This realization led us to consider looking at other features of the talks for our visualizations. After doing some brainstorming, we decided that topic extraction would be something that we would all find interesting, and we learned after fitting one of the 10 generated topics to each talk that the distribution of topics is much more spread out amongst the talks. Thus, we decided to move forward with topic extraction instead of the ratings provided from the TED site.

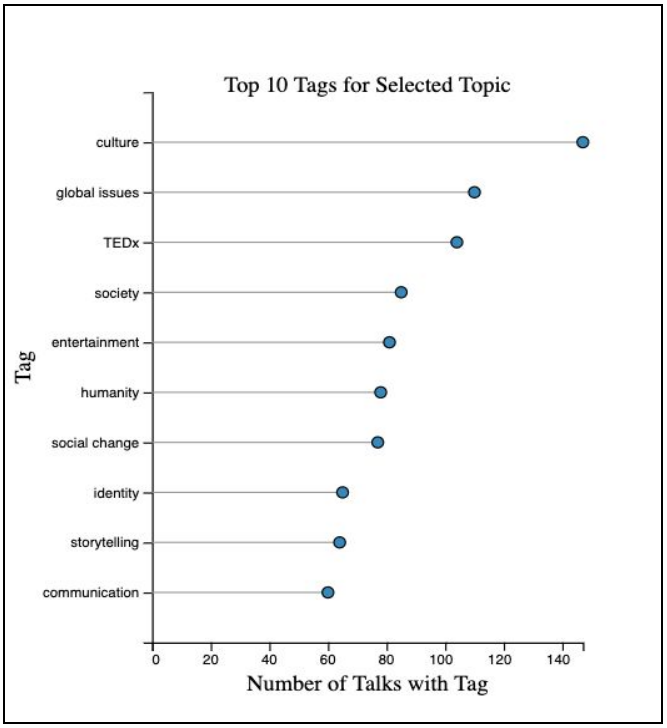
Implementation:

At a high level, our intended design will have three separate views whose displays will be affected by interaction on any of the graphs. There will be:

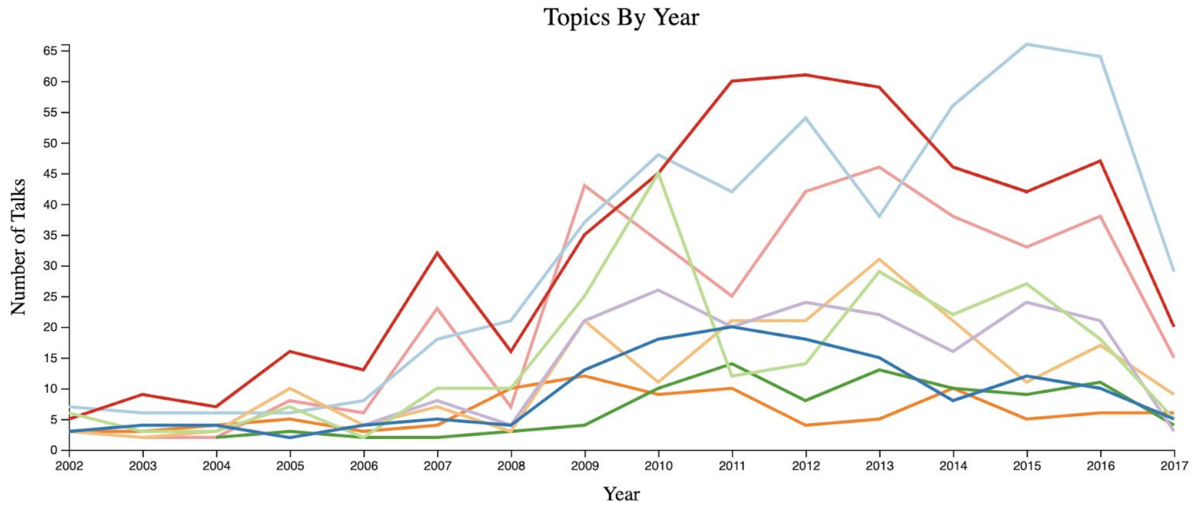
1. A stacked bar plot that shows, for each predicted group based on topic modelling, the overall proportion of ratings given by users of TED for videos in each category.



2. Initially, we were going to create a pie chart to show the most popular tags for a selected topic-model group. However, pie charts can be hard to interpret difference due to human difficulty in comparing angle, so we are going to move forward with a variation of a bar plot called a lollipop plot.

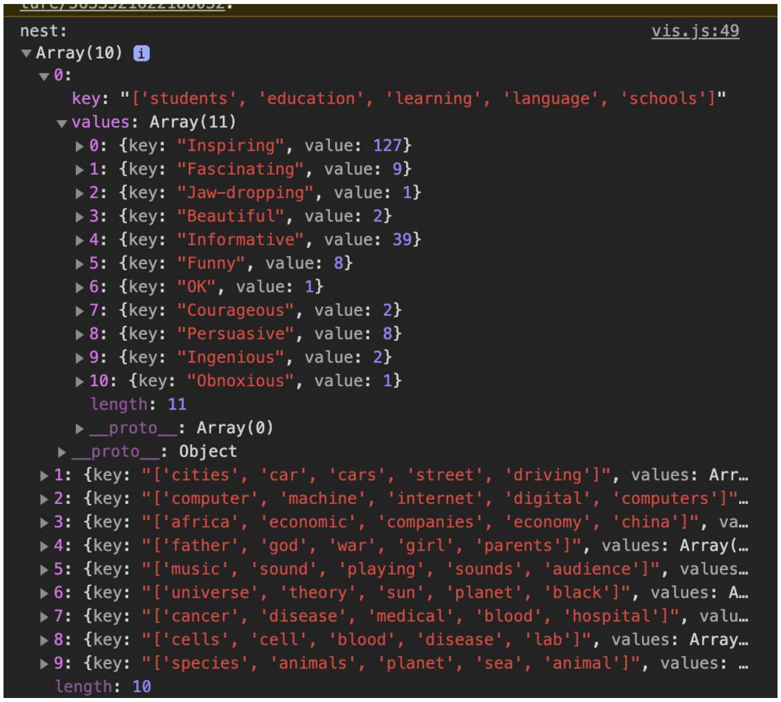


3. A line graph that shows the number of talks on a given topic from our model for each year. This allows us to demonstrate the changes in the content of Ted Talks over time.



Blockers:

Before creating a working prototype for the stacked bar plot, one must make sure that the data is in the correct structure to perform the most efficient data join possible. Right now, my data is in a nested structure like such:



```
nest:                                                            vis.js:49
▼Array(10) i
  ▼0:
     key: "['students', 'education', 'learning', 'language', 'schools']"
   ▼values: Array(11)
     ▶0: {key: "Inspiring", value: 127}
     ▶1: {key: "Fascinating", value: 9}
     ▶2: {key: "Jaw-dropping", value: 1}
     ▶3: {key: "Beautiful", value: 2}
     ▶4: {key: "Informative", value: 39}
     ▶5: {key: "Funny", value: 8}
     ▶6: {key: "OK", value: 1}
     ▶7: {key: "Courageous", value: 2}
     ▶8: {key: "Persuasive", value: 8}
     ▶9: {key: "Ingenious", value: 2}
     ▶10: {key: "Obnoxious", value: 1}
      length: 11
     ▶__proto__: Array(0)
   ▶__proto__: Object
  ▶1: {key: "['cities', 'car', 'cars', 'street', 'driving']", values: Arr…
  ▶2: {key: "['computer', 'machine', 'internet', 'digital', 'computers']"…
  ▶3: {key: "['africa', 'economic', 'companies', 'economy', 'china']", va…
  ▶4: {key: "['father', 'god', 'war', 'girl', 'parents']", values: Array(…
  ▶5: {key: "['music', 'sound', 'playing', 'sounds', 'audience']", values…
  ▶6: {key: "['universe', 'theory', 'sun', 'planet', 'black']", values: A…
  ▶7: {key: "['cancer', 'disease', 'medical', 'blood', 'hospital']", valu…
  ▶8: {key: "['cells', 'cell', 'blood', 'disease', 'lab']", values: Array…
  ▶9: {key: "['species', 'animals', 'planet', 'sea', 'animal']", values: …
   length: 10
```

This is the closest I've been able to structure the data to what I need in the stacked bar plot, but I will likely need to consult office hours to figure out the correct way to approach this.

Design Evolution:

Throughout our project, we considered a wide range of visualizations, which often changed as we reconsidered what questions we wanted to answer from our data set as well as what would lead to meaningful analysis.

Initially, we wanted to design a heat map placing tag values that TED provides on each axis and then encoding the number of videos that feature these two tags together with a color scale. We also wanted to design a line graph with each topic as a path showing the number of talks released for that topic each year. Finally, we wanted to include a graph grouping topics by their most popular tag. Within each grouping, we would then group each topic in that grouping by their second most popular tag and so on.

We ended up deviating from our proposal in a lot of ways. One of the main changes that we made was in deciding to use topic modeling to categorize the TED talks. In conducting our initial analysis of the data, we realized that there were far too many unique tags provided by TED and this made our visualizations difficult and often times meaningless. For example, when trying to visualize our line graph, we had difficulty with tags such as “God” and “Religion” appearing separately. Because these tags were provided by TED themselves, we were unable to ensure consistency in the data and how it was labeled. We also had access to the full transcripts, so we decided to use NMF topic modeling to label the talks ourselves (see Data section). This decision changed the way we decided to approach a lot of our other visualizations.

Our topic modeling resulted in 10 unique topics representing our entire data set. As a result, it no longer made sense to use the heat map, as these topics were distinct and each video fell into only one topic area. This meant that there was no longer an overlap between topics amongst our videos. Similarly, our graph grouping by topic did not seem like an appropriate visualization as we only had 10 unique topics. However, we did decide to keep our line graph.

After revisiting our designs, we decided to move forward with the line graph with a path for each topic showing the number of videos per year for that topic. This meant that we would only have 10 lines and would be able to easily encode with space, which is preferred. Additionally, an additional visual channel, slope, would help to show changes in the popularity of a given topic year to year.

We also decided to replace our heatmap idea with a lollipop plot to visualize the tags that TED labeled for a given topic that we identified. Our y-axis would contain the tags and our x-axis would contain the number of talks labeled with a given tag. We opted to use this visualization over a pie chart as mentioned in our implementation section, as humans can have difficulty determining the difference in angle between sections in a pie chart, whereas a lollipop plot, which encodes with length, allows for easier analysis. This visualization is intended to provide more in depth information as to what a given topic contains, based on labels provided by TED. Some of our topics identified by our model can be fairly broad, so this visualization will provide additional insight into those topics and will also be helpful for supporting interaction.

Finally, we decided to replace our graph idea with a stacked bar chart, which for each topic area, displays the proportion of ratings as provided by TED users. These ratings provide insight into the qualitative reactions that TED users had to each talk, ranging from reactions like “Inspiring” to “Informative”. We opted to use a stacked bar chart, as it allows us to encode the proportion of ratings with length, allowing for easy comparison between ratings for a given topic as well as across topics. This visualization helps us to understand how TED viewers perceive each topic, which was a goal of our project.

Analysis:

Our analysis is continuing to evolve as we further develop our visualizations. So far, we have seen that TED talks as a whole have increased in popularity, as the number of videos across nearly all topics has increased over time. We have also seen that the vast majority of talks are deemed “inspiring” and “informative”, which makes sense given the nature of TED talks. It is more difficult to summarize the tags by topic lollipop plot, but the analysis for each given topic area provides a lot of insight into what TED talks are about and what is included in the topic as labeled by our model.

As we continue to develop our visualization, we want to support more interaction between visualizations. For example, selecting a given topic area should update its tags by topic view as well as in some way highlight its topic in the line graph and the stacked bar chart. This will allow for more meaningful analysis on a topic level across our various visualizations.