Automatic Unsupervised Document Clustering and Labeling: A Human-Knowledge Based Approach

Juan S. Rodriguez, Bailey Smith
Stuff and stuff and stuff

March 31, 2018

Abstract

Document labeling allows users to understand the content of available information such as scientific papers, movies, books, music, legal documents, and any medium from which text can be extracted. Labeling is an inherently biased problem, since it is restricted by the labels that a human can provide. We suggest an innovative method that generates labels for documents in an unsupervised way. This implies no human suggestions for labeling, and thus no human selection bias. We demonstrate the usefulness of this algorithm by labeling TED talks using only their transcripts.

1 Introduction

1.1 Motivation

Our primary objective is to create a labeling and clustering method to process documents, videos, or talks based solely on their transcript. Our secondary objective is to eliminate the need for human intervention during this process. We will be using the transcripts on all the TED talks that have been published up to September 2017 (roughly 2400).

An automated classification method would allow better suggestions to the online audience, provide an **unbiased-human classification**, and would be generalizable to other fields and problems. This sort of classification method would give the tools for a machine to sort through thousands of legal documents, health documents, speech transcript, or books, and allow the scholar or reader to approach them seamlessly by following their labels.

We have found several research papers that have attempted to solve this problem. We suggest that our approach can propose a better genre or topic labeling for large document sets (e.g., hundreds of books, legal documents, or movies) or give a human-like description of a unique document (e.g., a book). The reson for this is that most labeling techniques rely on in-document information, which can be limited, or a trained or pre-determined labeling set. Our approach does not depend on the limitations of within-document content nor human chosen labels to tag the data.

1.2 TED Talks

Since 1984, TED has become an iconic conference in which experts around the world present their ideas and analysis in the fields of technology, entertainment and design (T.E.D.). Since 2006, the conference platform decided to make every talk public and free by publishing them on their

website. Given TED's history and prestige, TED talks have become a standard for quality when it comes to delivering an informational talk to an audience. We will use the transcripts that come from their talks to suggest individual talk labeling and cluster labeling. We will compare our labels with those already established by TED.

In this report, we will first give a thorough description of the data we used. Next we will describe the models and algorithms that we used in order to correctly label and classify the data. After that, we will explain the results of applying our method to the TED talk dataset. Finally, we will discuss how our method compares with the current way of labeling and clustering TED talks.

2 Data

2.1 Description

TED talks have very diverse content in their talks. For this reason we would like to provide some insight in what the data looks like. The most common TED speaker occupation is "Writer" with 45 occurrences, followed by "Designer" with a total of 34 occurrences. The total number of occupations among speakers is 1458. The average talk is 13.7 minutes long, with the shortest being 2.25 minutes and the longest being about 1.5 hours, which was given by the author of "The Hitchhiker's Guide to the Galaxy".

The average TED talk has been translated to 27 languages. There are 86 talks that have no assigned language because they are mainly musical presentations. These talks willnot be used as there is no transcript available for them. The average number of views per talk is 1.6 million and the talk with highest number of views has been seen almost 50 million times and it's called "Do schools kill creativity?".

With regards to the video tags, TED gives each video a number of possible tags to link a talk with different topics. The average video has 7.56 tags, with some videos having over 30 tags and some having just one. We will be comparing our topic labeling outcome with these tags. The other important variable that will be useful for us to observe is the related talks. Every video is given a connection to 6 other videos that are suggested for the viewer to watch. We will measure the accuracy of our bisecting k-means with the overlap it has with these suggested videos. We suggest to improve these two metrics (the tags and the related videos), by suggesting tags that are more useful for the viewer, and providing a more robust connection between "related talks" and the tagged labels.

2.1.1 Transcript Dataset

The transcript dataset contains the transcripts for 2467 TED talks. In this database we found three duplicates. We decided to analyze only the talks that are found on both databases in order to have homogeneous data. The 86 talks for which there is no transcript data are the music ones we had referred to before. We discarded any dupiclates.

2.2 Collected Variables

Table 1

Variable	Type	Description
title:	str	The title of the talk
description:	str	A blurb of what the talk is about
main_speaker:	str	The first named speaker of the talk
speaker_occ:	str	The occupation of the main speaker
duration:	int	The duration of the talk in seconds
url:	url	The URL of the talk
related_talks:	dict	List of dict of 6 related talks
tags:	list	The themes associated with the talk

Source: The data has been scraped from the official TED Website and is available under the Creative Commons License. It was retrieved from the Kaggle featured data sets in October 2017.

3 Methods

3.1 Algorithm Overview:

The process will be divided fundamentally into two different parts.

The first part will be the labeling process. The labelling process will be split into three different steps. The first step will be to apply a Latent Dirichlet Allocation (LDA) using a Gibbs Sampling technique in order to find the prevalent topics in the whole corpus of TED talks. This will give us a list of words for each topic which describes the topic. The second labeling step is to find a tagging label for each topic. Given the list of words assigned to each topic, we implement and algorithm to assign a macro concept that encapsulate all the words for each topic. For example, our LDA may have as an output the following list: `['government', 'party', 'elections', 'voting', 'candidate']`. Step two will label this list of topics under the concept `Politics`. Step three is applying these topic labels to the different clusters that will be provided by part two.

The second part will be a two-fold. First, we will use a bisecting K-Means algorithm in order to determine what the main clusters in the text are. Second, we will use a cosine similarity system to find related documents.

1. Creating Labels
    2. LDA - Prevalent Topics
    3. Wiki, Glove - Label the topics
    4. Labeling
        - Cluster Labeling
          -or-
        - Individual Topic Labeling
2. Clustering
    2. Bisecting KMeans - Main Clusters
        3. Count Vectorizer

4. sklearn.feature_extraction.text.CountVectorizer

3. Cosine Similarity - Related Talks

3.2 Algorithm Description:

3.2.1 LDA with Gibbs Sampling Topic selection

The Latent Dirichlet Allocation is a generative statistical model that assumes there exist hidden states that are represented to some extent in each document. The algorithm requires several documents (e.g., a set of TED talks, or even just the paragraphs in a text) and assumes that each document can be encapsulated under some of the hidden states, or in this case topics. We use the LDA to find the main topics in the transcripts and return a list of words describing the given documents. Table 2 contains a sample of the ten lists of words generated by the algorithm identifying ten hidden states, or topics, among the Ted Talks.

Table 2

Topic 1	school	learning	students	education	teachers
Topic 2	water	energy	earth	planet	climate
Topic 3	virus	hiv	disease	flu	malaria
Topic 4	people	human	social	life	love
Topic 5	language	books	laughter	english	words
Topic 6	ocean	fish	sea	boat	water
Topic 7	universe	light	space	stars	physics
Topic 8	city	car	urban	street	york
Topic 9	world	percent	money	dollars	africa
Topic 10	cancer	health	patients	cells	disease

3.2.2 Wiki and GloVe Topic Selection

Given the former data that was extracted through the LDA algorithm by means of a Gibbs Sampling method, we presented a method to label each of these lists of words that were given as output. To start, we identify the summary of each word in Wikipedia, we then proceed to clear out words that do not have substantive meaning (i.e., stop words) and we proceed to see what the overlap is between the words per topic. In a parallel approach, we use the "Global Vectors for Word Representation" (GloVe) created by Stanford in order to understand other possible training words.

(TO VERIFY) After using these two approaches, we have obtained a count for the overlapping words, and we then to choose the two with the most repetitions among the two criteria. We gave a higher weight to the Wikipedia approach, given that GloVe tends to contain digit representations that aren't semantically valuable (e.g., ZIP codes to identify a location). We use these words to provide labeling and that is represented in the following table with the examples given by the LDA.

Table 3

Topic 1	School Education	school	learning	students	education	teachers
Topic 2	Planet Supplies	water	energy	earth	planet	climate
Topic 3	Disease Symptoms	virus	hiv	disease	flu	malaria
Topic 4	Social Concept	people	human	social	life	love
Topic 5	Words Meaning	language	books	laughter	english	words
Topic 6	Earths Water	ocean	fish	sea	boat	water
Topic 7	Space Scientific	universe	light	space	stars	physics
Topic 8	City Transportation	city	car	urban	street	york
Topic 9	States Fact	world	percent	money	dollars	africa
Topic 10	Health Care	cancer	health	patients	cells	disease

3.2.3 Bisecting K-Means

While reviewing research that has been done in the past, we stumbled upon a comparison study that compares performance and efficiency of different text clustering techniques. The paper highlighted that hierarchical algorithms usually perform a better classification that other algorithms. They conclude the following:

> "'bisecting' K-means, can produce clusters of documents that are better than those produced by "regular" K-means and as good or better than those produced by agglomerative hierarchical clustering techniques"

This is specially valuable given that most hierarchical clustering techniques have at least a quadratic complexity. On the other hand, k-means has a linear complexity. Therefore, this approach is not only very accurate, but it also is very cheap when it comes to computational resources.

The Bisecting K-Means algorithm begins by splitting the data into two clusters using the K-Means algorithm. The algorithm then takes each centroid and finds the distance between all the points in the corresponding cluster. The cluster with the largest total distance is then chosen to be split into two new cluster by the K-Means algorithm. This process is continued until the data is divided into the desired number of clusters. (Explain why this method was chosen for our data/our problem)

Data: K: Number of clusters
X: List of Documents
vectorize(X);
clusters, Centers = k-means(X,k=2);
list = dist(clusters,centers;
for *i in 1,2,...,k-1* **do**
  index = argmax(list);
  clusters, Centers = k-means(X[index],k=2);
  list = dist(clusters,centers);
end
Result: K Clusters

Algorithm 1: BisectingKMeans Clustering

3.2.4 Cluster Labeling

After using the bisecting k-means algorithm to obtain k clusters, we will go through each one of these clusters and aggregate the text of all the documents contained in them. Therefore we will have k large documents that we will call macro-documents. Afterwards, we will create a vector representation each of these macro-documents and we will use a similarity measure **(we will test cosine similarity, Jaccard similarity, and a L_2 norm)** to determine which topics are prevalent within each one of the macro-documents. This will allow independent labeling for each one of the clusters.

To illustrate this, let's assume we generated 7 clusters and have aggregated them into seven macro-documents. We know that three of the topics identified by the LDA are *Technology Innovation*, *Genetic Biology*, and *City Transportation*. Now lets assume that their is a macro-document **X** that contains mostly innovation in biology and genetics TED talks, and another macro-document **Y** that contains technology and innovation in urban transportation. These two clusters contain information about technology and innovation in two different manners. Therefore **X** will receive the tags *Technology Innovation* and *Genetic Biology*, and **Y** will receive the tags *Technology Innovation* and *City Transportation*. After the topics have been assigned to each macro-document, we proceed to assign those labels to the documents that originally corresponded to the cluster. In these manner the topics are non-exclusive between clusters, and granularity is controlled by the number of clusters desired, which is variant from case to case.

TO DO: what would happen if we use a sampling method to draw the cluster labels?

3.2.5 Individual Document Labeling

In a similar fashion, we can use the approach done to sets of documents to an individual document. We assume an individual document will be shorter and that the document has subdivisions like paragraphs or chapters. For sake of simplicity we assume that it is a chapter. We then treat every chapter as a document and proceed to draw the main topics within the text with LDA.

After having the topics, we use our topic-labeling approach and get the main labels for the total individual document. Then we proceed to do the bisecting k-means using the chapters and proceed the rest of the algorithm as explained before. All the main topics extracted by the LDA would allow to create a summary of the text, and the cluster labeling would allow to know what chapters talk about what topics. This would be a great tool for legal documents, when a lawyer or judge is looking for important information within a book or case file. The algorithm would create a general summary of the file and a detailed index of where to go for specific information.

4 Results

We applied the algorithm to the TED talk transcripts dataset, and we obtained the results shown previously to illustrate the algorithm. By using 21 topics and a 300 cluster approach and a 7 cluster approach we got this interesting results. (We will visualize the 300 cluster result to compare with the "related talks" variable in order to measure the usefulness of the bisecting k-means).

300 Clusters: (**TODO** We will visualize the 300 cluster result to compare with the "related talks" variable in order to measure the usefulness of the bisecting k-means) *Seven Clusters:* Each cluster was labeled as follows: (**TODO** Show the labels and some titles found in the clusters. Visualize the 7 cluster approach with the "tags" TED already has and compare them to our tags to see similarities.)

The suggestions found on the website do this: (Compare a sample of ten talks labeled by the algorithm vs by TED)

5 Analysis

Based on the comparison between the labels we found and original labels: (whether or not the topics found using our methods are as good or better than the human chosen topics) Our hope is that Our methods will prove to enhance the labeling of TED talks into topics in order to improve user experience.

Based on the comparison between the clusters we found and the "related talks": (whether or not the clusters found using our methods are as good or better than the human chosen "related talks") Our hope is that Our methods will prove to enhance the clustering of TED talks by topics in order to improve user experience.

6 Conclusion

In this research, we presented a method of classifying documents based on the words contained in it. We first abstracted from these words to create groups that indicate what was talked about in the document with LDA. Next we used Wikipedia and GloVe to find a topic that described each group of words. Then we grouped the documents together using Bisecting K-Means. After that, we matched the topics to the clusters of documents which gave us the final result. This method is unsupervised and thus eliminates human bias in choosing how to cluster and label TED talks into separate topics.

As the results suggest: (**TODO** summarize analysis).

In the process of developing the method, we realized that there may be a better way to implement the Bisecting K-Means algorithm that would identify clusters accurately. Further research should be done in this area to ensure that the separation of the TED talks is done in a way that will be helpful to viewers.