Clustering Wikipedia Articles

Lane Aasen
Department of Computer Science
University of Washington
Seattle, WA 98105
aaasen@cs.washington.edu

Abstract

Clustering Wikipedia articles using K-Means. Exploration into K-Means including choice of K, K-Means++, the effect of random seeds, and tf-idf thresholds.

1 Dataset

The provided dataset contains 15,903 Wikipedia articles in term frequency inverse document frequency (tf-idf) format. Most of the articles are in English, but some are not. Sorting the documents by language would be an interesting project in itself! There are 10,574 unique words in this dataset. Each document is represented as a sparse vector with one dimension for each word. Stop words have been removed from the dataset, but uncommon words remain.

2 K-Means Clustering

My first goal was to implement a basic K-Means clustering algorithm from scratch. The code is available at `https://github.com/aaasen/wiki-cluster`.

2.1 Choosing K

2.1.1 Minimizing Distortion

Given K clusters $C_1, C_2, ..., C_K$ where each cluster is a set of document vectors and μ_i is the centroid of C_i, the total distortion is defined as follows:

$$\sum_{i=1}^{K} \sum_{d \in C_i} ||d - \mu_i||^2$$

To minimize the distortion, we could set K equal to the number of documents, but then the clusters would be meaningless. We want to choose a K with low distortion that also results in interpretable clusters. Figure 1 shows a plot of K versus total distortion. When $1 \leq K \leq 16$, adding additional clusters has a large impact on the distortion, but once $K > 16$, adding additional clusters has little impact on the distortion. From this alone, it makes sense to set $K = 16$ since it provides a good balance of distortion and interpretability.

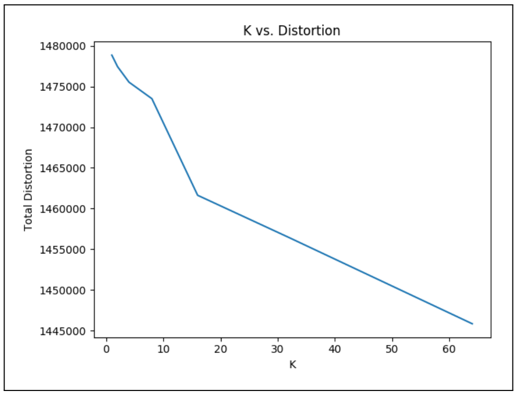

Figure 1: K versus total distortion for $K \in \{1, 2, 4, ..., 256\}$

2.2 K and Cluster Size

As K increases, the clusters become more sparse. Once $K = 256$, over half of the clusters have only one document, and are essentially useless. When $K = 16$, the median cluster size is 8.5, and the cluster sizes are as follows:

$$[10061, 3013, 1128, 909, 707, 30, 23, 13, 4, 4, 3, 2, 2, 2, 1, 1]$$

Over half of the clusters are very small, and one of the clusters is too large to be interpretable. This indicates that the data has significant outliers and may lack a structure conducive to clustering.

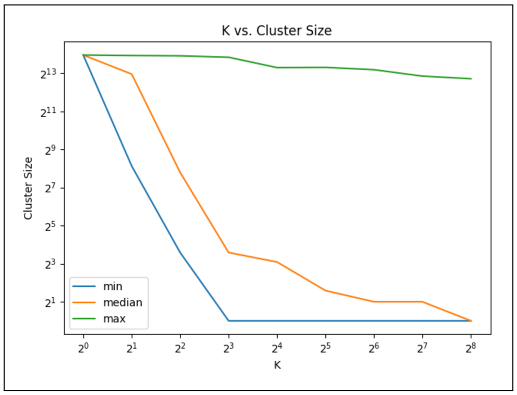

Figure 2: K versus minimum, median, and maximum cluster size for $K \in \{1, 2, 4, ..., 256\}$ with a log_2 scale on both axes.

2.3 Exploring Clusters

Table 1 shows the clusters with at least 10 documents for K-Means clustering with $K = 16$. The words in each cluster are the dimensions of the centroid with the largest magnitude. The documents shown are those that are closest to the centroid of the cluster.

Overall, the generated clusters make sense, but there are some points of confusion:

- The words that make up cluster 0 have little relation to each other. This cluster contains the majority of the documents.
- Cluster 1 contains churches as well as colleges.

- Cluster 3 contains documents related to TV shows and sports because both contain the word ”season.” Another explanation for this is that sports players appear on Dancing with the Stars.

Table 1: K-Means clusters with $K = 16$ and at least 10 documents.

Cluster	Size	Words	Documents
0	10061	females station family located north	mcgillpainquestionnaire historyofthefamily thetussaudsgroup nadiraactress mansfieldsummithighschool
1	3013	church college students published institute	edmondscommunitycollege helderbergcollege oberlincongregationalchurch lundbyoldchurch dioceseoflimerickandkillaloe
2	1128	party served general member senate	partyidentification labourfarmerparty democraticalliancesouthafrica liberaldemocratsitaly christiancreditparty
3	909	season club playing seasons player	dancingwiththestars davidmccracken gilbertcurgenven bjsamsamericanfootball livingstonewalker
4	707	album released songs records rock	thegreatestdaytakethatalbum conflictingemotions primalscream leftbacklp elisamartin
5	30	nba basketball points season seasons	kcjones hakeemolajuwon albertkingbasketball ballstatecardinalsmensbasketball 201011southfloridabullsmensbasketballteam
6	23	riots police murder captured robbery	sowetouprising 1992losangelesriots nikolaybogolepov josephlamothe jenmi
7	13	congo subtropical republic zambia zimbabwe	republicofcabinda brownrumpedbunting copperbeltprovince leptopelisviridis yellowthroatedpetronia

2.4 K-Means++

K-Means++ is a method for selecting the initial cluster centers for K-Means. In the normal version of K-Means, initial cluster centroids are selected by sampling random documents from the dataset. This can produce suboptimal clusters, increase time to convergence, and increase variability in clustering performance.

K-Means++ solves this by selecting initial centroids one at a time in order to minimize distortion. Each document is given a weight proportional to the distance to the nearest centroid and the next centroid is selected at random using these weights until K centroids have been chosen. This results in better centroids than random sampling and lessens the time until convergence. However, it can be quite expensive when K is large since the distance from each document to each cluster centroid must be computed at every iteration.

I implemented K-Means++ and expected it to increase clustering performance considerably. However, I found that it actually had a slightly negative impact on distortion and noticeably increased cluster sparsity.

I think that this is because outliers were chosen as initial centroids and the clusters never expanded to include other points.

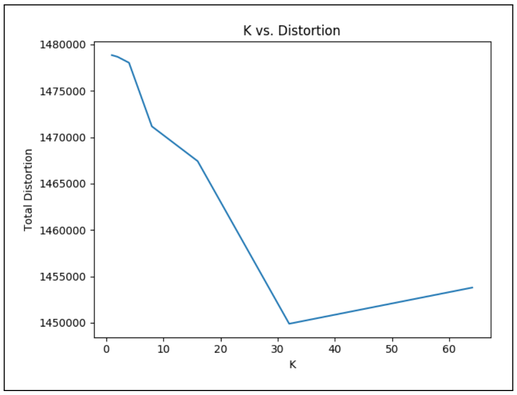

Figure 3: K versus total distortion for $K \in \{1, 2, 4, ..., 64\}$ using K-Means++.

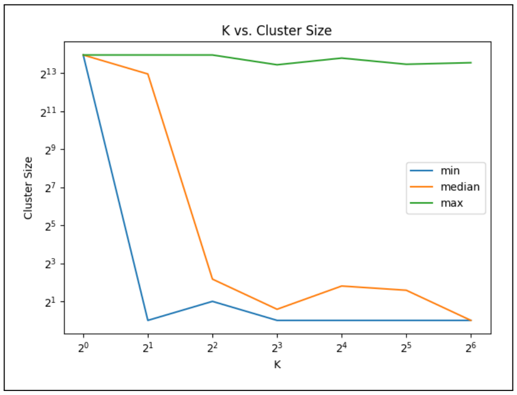

Figure 4: K versus minimum, median, and maximum cluster size for $K \in \{1, 2, 4, ..., 64\}$ with a log_2 scale on both axes. Using K-Means++.

2.5 Fighting Cluster Sparsity

After implementing K-Means++ with no success, it became apparent that there was a problem with the underlying data, and not with the clustering algorithm.

I began to investigate the provided dataset, which was in tf-idf format with no documentation. I found that stop words had been removed from the dataset, but rare words had been left in. Half of the words in the dictionary were used in 16 or fewer datasets (less than 0.1%). Tables 2, 3, and 4 show the least common, somewhat common, and most common words in the dataset, respectively.

2.5.1 Selecting a Threshold

Since the provided dataset does not contain stop words, I decided not to filter out common words. The most common words in the dataset appear to contain useful information for clustering.

Table 2: 10 least common words with the number and percentage of documents that they appear in.

Word	Documents	Percentage of Documents
frazioni	0	0.0%
threeletter	1	0.0062881%
budjovice	1	0.0062881%
gmina	1	0.0062881%
ortsgemeinden	1	0.0062881%
headwater	2	0.012576%
baronetage	3	0.018864%
breaststroke	3	0.018864%
voronezh	3	0.018864%
rosettes	3	0.018864%

Table 3: 10 words near the median with the number and percentage of documents that they appear in.

Word	Documents	Percentage of Documents
multimillion	16	0.10061%
pisa	16	0.10061%
pranks	16	0.10061%
pesticides	16	0.10061%
hesitation	16	0.10061%
convection	16	0.10061%
ortiz	16	0.10061%
stagnation	16	0.10061%
gonzlez	16	0.10061%
cummins	16	0.10061%

Table 4: 10 most common words with the number and percentage of documents that they appear in.

Word	Documents	Percentage of Documents
well	4009	25.209%
second	3538	22.247%
high	2836	17.833%
family	2579	16.217%
group	2412	15.167%
north	2364	14.865%
major	2298	14.45%
large	2227	14.004%
general	2187	13.752%
long	2164	13.607%

Selecting a lower threshold involves finding the right balance between information loss and cluster sparsity. Words that are only used in one document should be ignored since they provide no useful clustering information. The problem gets murkier with words that are used in just a few datasets. Table 3 lists ten words that are somewhat common, in that their counts are exactly at the median word count. Some of these words seem useful for clustering, such as ”multimillion”, ”pesticides”, and ”convection”. Others are names which do not necessarily indicate document similarity.

One of the large problems here is that we are working with only 15,000 of the 5 million English Wikipedia articles. If we were working with more data, we could produce more fine-grained clusters with rarer words. However, since we only have a tiny subset of Wikipedia, we just don’t have enough examples to produce these narrow clusters.

There is no correct way to choose a tf-idf threshold. I ended up choosing to get rid of the least common half of words, but many other choices would be just as valid.

2.6 Effect of Seed on Distortion

After observing some unexpected variability in my results, I decided to investigate the effect of random seeds on distortion.

I found that the choice of seed can have a large impact on cluster quality and distortion. Because of this, it is very important to run K-Means with at least a few different seeds and select the best one. While this can be too computationally expensive to do when running experiments, it is important to do when generating the final clusters.

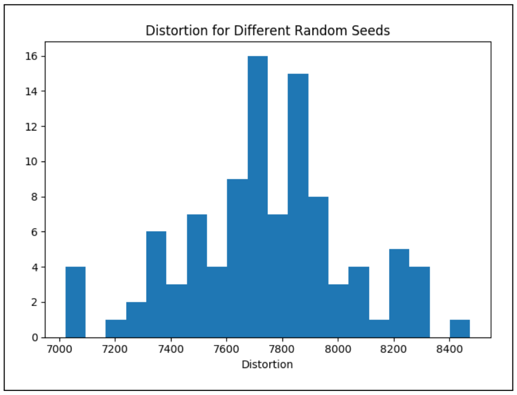

Figure 5: Distortion using K-Means++ on 100 training documents with different random seeds.

3 Conclusion

Table 5 shows the final clusters. While 64 clusters were generated, only the largest 10 are shown. Only words that occurred in at least 16 documents, which is about half of all words, were considered. K-Means++ was used to generate the initial cluster centroids. Names have been added to each cluster.

3.1 Issues with K-Means

3.1.1 Variability in Cluster Size

The most glaring issue with these results is that the sizes of the clusters vary greatly. One cluster contains about half of the documents while the other 63 contain the other half. There are many useless one document clusters.

It is difficult to know if this is an intrinsic part of the data, or an issue with the K-Means algorithm. I suspect that it is a combination of both, and that K-Means is not the best model for the data.

While exploring the data, I did try to visualize it using Multidimensional Scaling (MDS), but I found that too much information was lost in the dimensionality reduction for it to be of any use. Textual data is high dimensional and sparse by its nature, which makes it somewhat unintuitive to work with in this context. It would be interesting to further explore dimensionality reduction techniques as well as alternate clustering algorithms and see if they could help address the problem of cluster sparsity.

3.1.2 Different Meanings of the Same Word

Some clusters contain documents that are related by a single word, where each document refers to a different meaning of the word. For example, cluster 9 contains documents about mars. Its main words are ”mars”, ”crater”, ”astronomical”, ”expedition”, and ”planets”. However, there are two documents in this cluster that have nothing to do with mars. ”hostagefilm” (*Hostage* (film)) is an

American action movie whose main character is named Marshall ”Mars” Krupcheck. ”marsrapper” (*Mars* (rapper)) is a Bay Area rapper who goes by ”Mars”.

3.2 Further Research

I had intended to explore Latent Dirichlet Allocation but ended up focusing on K-Means instead. As a starting point, I used SciKit’s LDA implementation, which seemed to work quite well. The words in each topic are related just as they should be. This would have been be a great topic to research more, if I had the time.

Table 5: 16 topics generated using Latent Dirichlet Allocation.

Topics
magazine, published, journal, format, awards
painting, gallery, paintings, painted, station
greek, empire, ancient, jews, dynasty
served, appointed, general, senate, member
systems, system, design, device, czech
party, security, elections, rights, police
station, air, route, railway, traffic
season, females, club, miles, tournament
church, building, house, located, castle
group, swedish, users, stockholm, program
engine, cars, comics, ford, novels
students, college, institute, campus, program
player, characters, red, color, blue
album, released, songs, musical, records
human, family, published, man, god
common, temperature, chemical, protein, oil

Table 6: 10 largest K-Means clusters with $K = 64$. Using words that appear in at least 16 (0.1%) documents with K-Means++ for initializing cluster centroids.

Cluster	Size	Words	Documents	Label
0	8486	females station family north located	kotavamsa tornadoesof1993 whitby colinmeads mladeniiiubiofbribir	
1	2226	students system high program technology	edmondscommunitycollege helderbergcollege stargateschool miltonhighschoolmiltongeorgia govthazimuhammadmohsincollege	Colleges
2	2085	church published daughter royal paris	molire oberlincongregationalchurch lundbyoldchurch stmaryschurchgrodno dioceseoflimerickandkillaloe	Churches
3	1001	party served member general senate	partyidentification labourfarmerparty serbianliberalparty bronwenmaher democraticalliancesouthafrica	Political Parties
4	932	season club playing seasons player	bjsamsamericanfootball dancingwiththestars davidmccracken johnflanaganfootballer gilbertcurgenven	Sports
5	640	album released songs records rock	primalscream thegreatestdaytakethatalbum conflictingemotions bornthisway sweetkisses	Rock Music
6	280	japanese japan chinese pearl characters	frederickringer astorhousehotelshanghai19221959 japanesebadger imperialjapanesearmyairforce listofflclepisodes	Imperial Japan
7	44	pop songs album chart rock	popmusic teenpop britishpopmusic talkinginyoursleepcrystalgaylesong blahblahblahalbum	Pop Music
8	27	investigation money system june doctor	digitalmonetarytrust andyhayman martensvillesatanicsexscandal johnlittlechild unitedstatesvlibby	Crime
9	22	mars crater astronomical expedition planets	hostagefilm marsrapper entomopter mars2 carbonatesonmars	Mars

References

[1] MacQueen, J. B. (1967). Some Methods for classification and Analysis of Multivariate Observations. *Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability* . 1. University of California Press. pp. 281297. MR 0214227. Zbl 0214.46201. Retrieved 2009-04-07.

[2] Arthur, D.; Vassilvitskii, S. (2007). ”k-means++: the advantages of careful seeding” (PDF). *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms.* Society for Industrial and Applied Mathematics Philadelphia, PA, USA. pp. 10271035.