Eric Atkinson
Kabir Chandrasekher
Joseph Moghadam
Evan Ye

Clustering Wikipedia Walks

Introduction:
A well-known and curious fact about Wikipedia is that if a user starts on an arbitrary page, follows its first link, then follows the first link on the resulting article, and repeats, that user will with high probability eventually end up at the article for philosophy (this occurs for roughly 94.52% of starting articles, according to Wikipedia themselves[1]). Continuing after, that user will find themselves in a short loop of pages before returning to philosophy (6 pages as of 22 October 2014). If we consider only the first link, then, the pages explored converges to the same distribution for most start pages. In this project, we were curious to investigate to what degree this phenomenon might generalize to the rest of the links in a Wikipedia article. If we follow links randomly starting from different pages, will we tend to end up at the same locations? More formally, we consider the distribution of pages visited starting from different pages, and ask whether they converge to a single distribution, or clusters of distributions.

What we did:
1. Naive wikipedia crawler (src/crawler.py)
2. Page rank wikipedia crawler (src/PageRankCrawler.py)
3. Bhattacharyya distribution distance/K-L divergence (src/bhattacharyya.py)
4. Visualization (src/visualize.py)

Naive crawler:
The naive crawler was a modified version of the EECS website crawler, modified to traverse wikipedia and filter out junk URLs. It was effective at producing quick and dirty results. Restricted to crawling 2000 pages to produce distributions, it crawled 150+ different starting locations, producing page distributions for each start location.

A quick glance at the results (results/data.txt) is a good qualitative basis for wikipedia being strongly connected. With the page on “United States” appearing in many of the random walks with high frequency, it’s a good hypothesis that Wikipedia is pretty tightly connected, with countries as the bridge between many different domains of knowledge represented on wikipedia.

Page rank crawler:
The page rank crawler was a further modified version of the naive crawler. It used the same random surfer model as the naive crawler. Instead of ranking pages naively based on the

[1] http://en.wikipedia.org/wiki/Wikipedia:Getting_to_Philosophy

number of visits, it ranked a page according to the following formula:

$$PageRank(A) = \sum_{i \in links} \frac{PageRank(i)}{Links(i)}$$

The initial rank of each page is uniform on the number of wikipedia pages (~1/4,000,000). Unfortunately, due to the substantially larger number of requests that had to be made in comparison to the naive crawler, no meaningful data was able to be collected.

Bhattacharyya distance/K-L Divergence Analysis:

The distributions we created were discrete distributions that were formed by taking the highest ranking nodes in a run and normalizing their values. Note that this normalization is not necessary in the pagerank model as the rankings already constitute a valid distribution. Had there been the computing power to get distributions from this, it would have been interesting to see the results of distances between the naive distribution and the pagerank distributions. We also implemented different methods of computing the differences in distributions; namely the Kullback-Leibler divergence and Bhattacharyya distance. We used these metrics to get a feel of an average distance between the distributions we found. Our averages ended up being 1.3707947714 for the Bhattacharyya distance and 0.0575382595096 for the Kullback-Leibler divergence. The K-L divergence seems to indicate a very small average distance between the distributions, however we must note that for many of the pairwise distributions we found, the K-L divergence is in fact invalid as the two sample spaces are different, or there are events with positive probability in certain distributions that do not appear in other distributions.

Visualization:

We attempted to visualize our results in two ways:

1) We treat the distribution of pages visited for each start page as a vector, and then plot the vectors.
2) We create a distance matrix of Bhattacharyya distance for each start page, and generate a plot of points from the distance matrix.

Details follow:

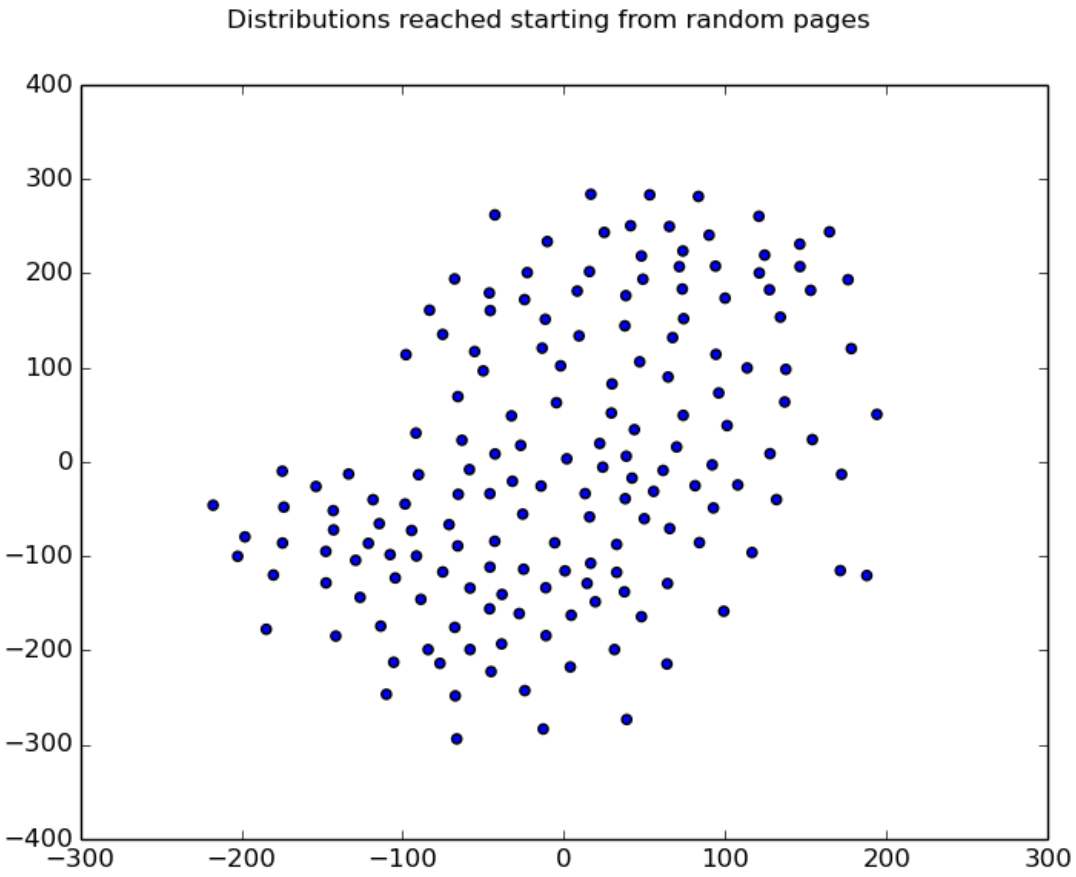
Each distribution vector corresponds to a particular start page. In a vector, each entry corresponds to a Wikipedia article that any one of the random walks visited, and each entry's value is the number of times the random walk from a particular start page visited to page. With 168 different start pages that each walked for 2000 steps before giving up, this turned out to be a very sparse vector.

We proceeded to visualize these vectors using t-SNE[2] , an algorithm for visualizing high dimensional data. t-SNE takes high dimensional vectors and reduces them down to a lower

[2] http://homepage.tudelft.nl/19j49/t-SNE.html

number of dimensions, usually two, so that they can be plotted and visualized. t-SNE's criteria for reducing dimensionality can be thought of in some sense as the opposite of PCA: whereas PCA tries to maintain as much variation in data as possible when it reduces dimensionality, t-SNE tries to maintain closeness in data; hence, it is well-suited for attempting to visualize clusters.

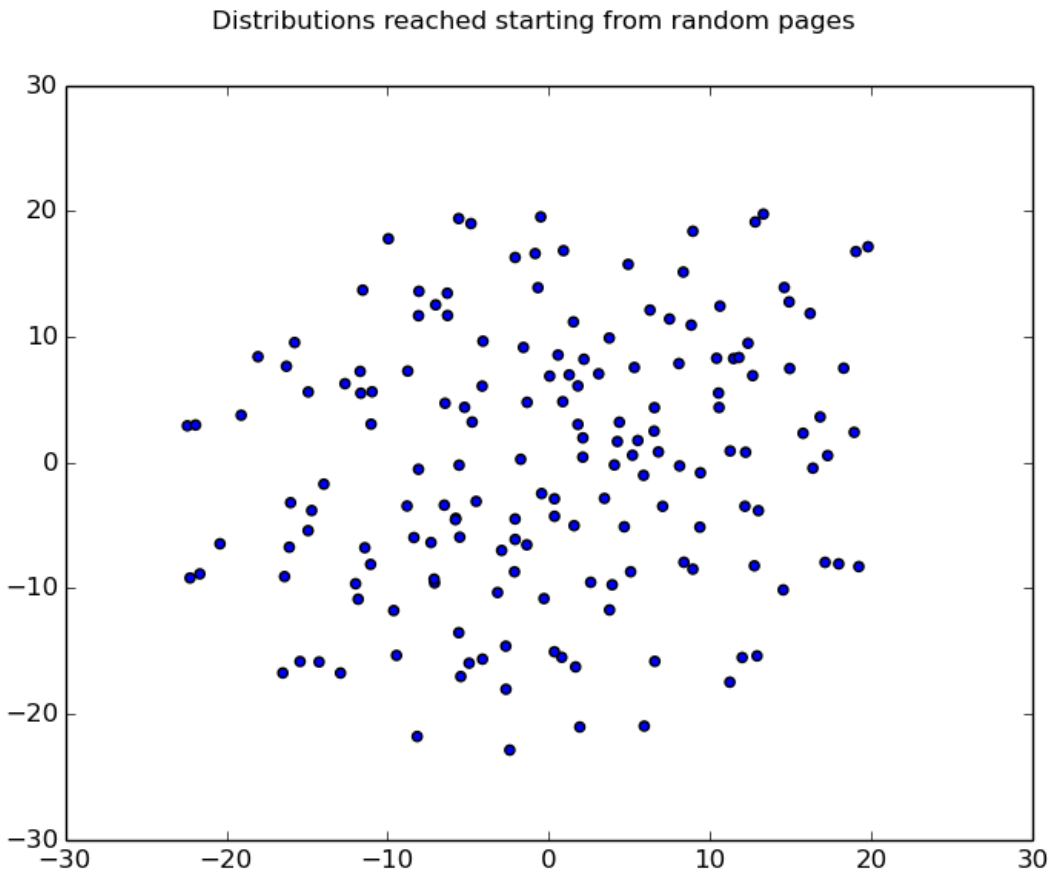
Unfortunately, our visualization of the vectors did not turn out to be very interesting:



(Note: the axes are essentially meaningless; this is nothing more than a 2D projection of high dimensional data intended to maintain closeness in 2D space if the vectors were close in the high dimensional space)

The distribution of points is very uniform, and does not reveal clusters we hoped to find.

We wondered if the sparseness of our vector was causing us trouble, so we next tried to visualize using the Bhattacharyya distance. We also used t-SNE for this, since it has functionality to derive a plot of points given only a distance matrix. This plot as well, however, did not appear to be very interesting:



At best we can say there are perhaps mini clusters beginning to form, since there are pairs of points here and there that seem close. However, it is not very strong evidence.

Conclusions:

From our results above, we cannot conclude that random walks from different start pages tend to result in visiting the same distribution of pages, nor can we conclude that they even generate clusters of similar distributions. However, even though these early results are not promising, we do not think the entire concept is entirely to be given up on. We suspect that, if there is structure to be discovered in Wikipedia, there is so much variance that the small amount of data we have is not nearly large enough to capture it. We perhaps need much longer walks (in the millions of lengths instead of the thousands), and more data points (maybe in the thousands) before patterns emerge if there are any. Unfortunately, since the small amount of data we have already took us several hours to collect, it was simply not possible to get that scale of data. It may, however, be a problem worth revisiting if we find ourselves with more computing power.