

CDS
Cornell Data Science

Text Mining and Graphs

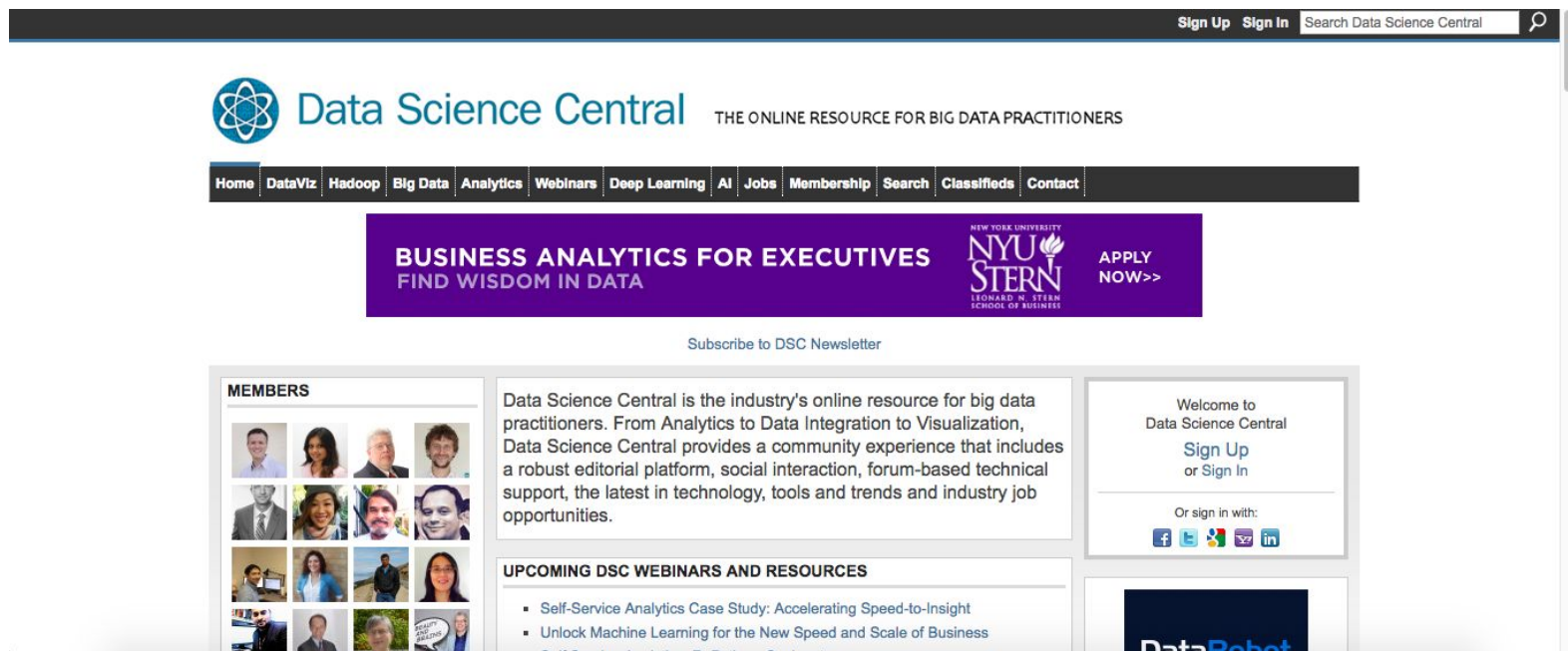


Real-World Application

Suppose you're making a student-led course and are figuring out what topics to include...



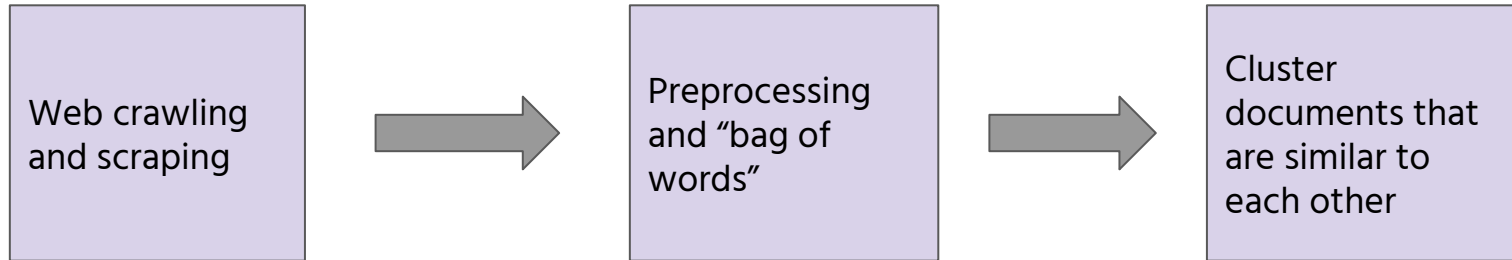
...but all you have at your disposal is a single website.



The screenshot shows the homepage of Data Science Central. At the top right, there are links for "Sign Up", "Sign In", and a search bar labeled "Search Data Science Central". The main header features the Data Science Central logo (a blue atom symbol) and the text "Data Science Central THE ONLINE RESOURCE FOR BIG DATA PRACTITIONERS". Below the header is a navigation menu with links: Home, DataViz, Hadoop, Big Data, Analytics, Webinars, Deep Learning, AI, Jobs, Membership, Search, Classifieds, and Contact. A prominent purple banner advertises "BUSINESS ANALYTICS FOR EXECUTIVES FIND WISDOM IN DATA" with the NYU Stern logo and an "APPLY NOW>>" button. Below the banner is a "Subscribe to DSC Newsletter" link. The main content area is divided into three columns. The left column, titled "MEMBERS", displays a grid of 12 member profile pictures. The middle column contains a paragraph: "Data Science Central is the industry's online resource for big data practitioners. From Analytics to Data Integration to Visualization, Data Science Central provides a community experience that includes a robust editorial platform, social interaction, forum-based technical support, the latest in technology, tools and trends and industry job opportunities." The right column, titled "UPCOMING DSC WEBINARS AND RESOURCES", lists two items: "Self-Service Analytics Case Study: Accelerating Speed-to-Insight" and "Unlock Machine Learning for the New Speed and Scale of Business". To the right of the main content is a sidebar with a "Welcome to Data Science Central" message, "Sign Up or Sign In" links, and social media icons for Facebook, Twitter, Google+, and LinkedIn. At the bottom right, a "DataRobot" logo is partially visible.

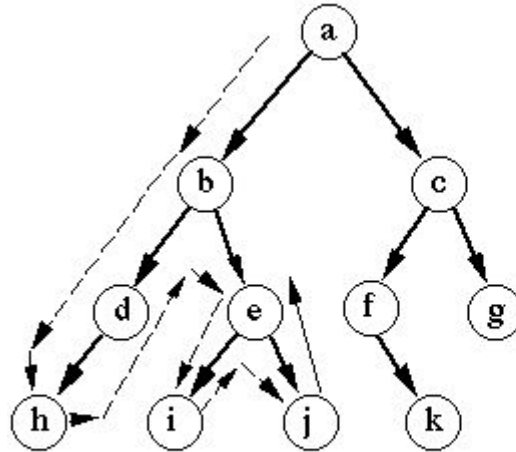


The Approach



Crawling

Web crawling - depth-first search of a set of linked pages. Google uses web crawling to build its search database.



Scraping

Scraping - Converting raw web pages into a usable format.

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>Example</title>
5     <link rel="stylesheet" href="style.css">
6   </head>
7   <body>
8     <h1>
9       <a href="/">Header</a>
10    </h1>
11    <nav>
12      <a href="one/">One</a>
13      <a href="two/">Two</a>
14      <a href="three/">Three</a>
15    </nav>
```



(Be nice and follow the rules of whichever site you're scraping!)

[Source](#)

Processing Text

Our goal: the **bag of words** format. Each document is converted into the a set of word frequency counts.

```
{“data”: 152,  
“science”: 138,  
“information”: 99,  
“learning”: 89,  
“machine”: 85,  
“model”: 70,  
“build”: 68}
```

Pros:

- Easy to interpret
- Space-efficient
- Can be put into matrix form

Cons:

- Ignores word order
- May not accurately encode meaning



Text Processing Techniques

“Cornell Data Science is an engineering project team at Cornell that seeks to prepare students for a career in data science.”

Punctuation removal/Lowercasing

“cornell data science is an engineering project team at cornell that seeks to prepare students for a career in data science”



Text Processing Techniques

Stop word removal

“cornell data science engineering project team cornell seeks prepare students career data science”

Lemmatization

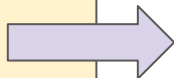
“cornell data science engineering project team cornell seek prepare student career data science”



Documents and Matrices

Here is what our document looks like now:

```
{“cornell”: 2,  
“data”: 2,  
“science”: 2,  
“engineering”: 1,  
“project”: 1, ...}
```



Document-term matrix

```
[2 2 1 1 1 1 0 0 1 ...]  
[0 1 2 1 5 1 3 2 3 ...]  
[1 1 9 1 8 0 0 0 10 ..]  
...
```



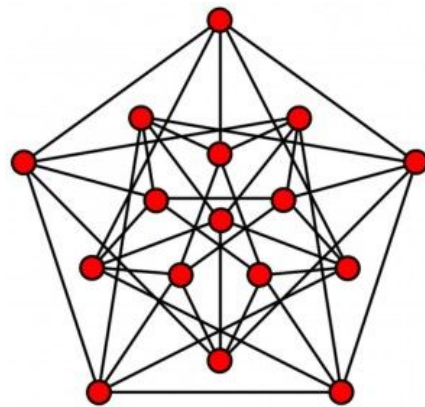
A Network of Documents

We build a **graph** of documents, where:

- Vertices (nodes) are documents
- Two documents are linked by an edge if they have a certain **cosine similarity**

This graph is **undirected**

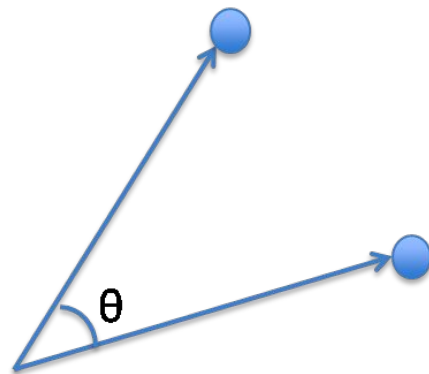
- If X links to Y, Y links to X



Cosine Similarity

Take the two rows (these are vectors) and find the cosine of the angle between the vectors.

$$\text{sim}(A, B) = \cos(\theta) = \frac{A \cdot B}{\|A\| \|B\|}$$



Why cosine similarity? Efficiently computed on sparse matrices. Also controls for length of vectors.

Building the Network

We take the document-term matrix, which is $d \times t$, and create a new $d \times d$ matrix. This is the **adjacency matrix** of the graph.

Document-term matrix

```
[2 2 1 1 1 1 0 0 1 ...]
[0 1 2 1 5 1 3 2 3 ...]
[1 1 9 1 8 0 0 0 10 ..]
...
```

Adjacency matrix

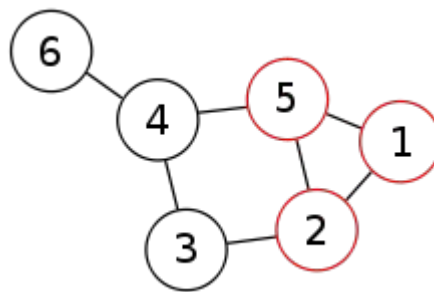
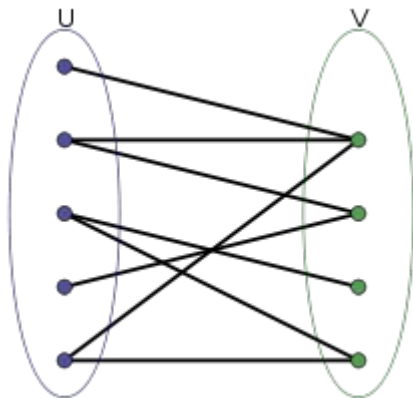
```
[0 0 0 1 1]
[0 0 0 1 0]
[0 0 0 0 0]
[1 1 0 0 1]
[1 0 0 1 0]
```



Graph Structures

Bipartition (bipartite graph) - two sets of vertices that are not internally connected but connect to each other

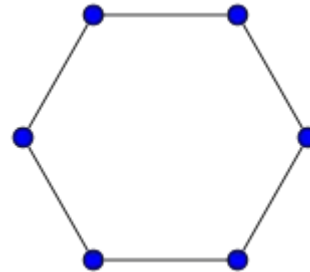
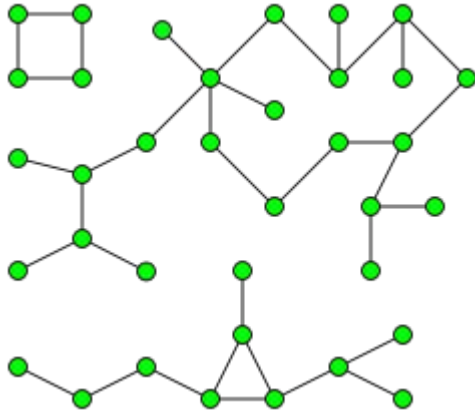
Clique - set of vertices that are all connected to each other



Graph Structures

Connected component - maximum-size sets of vertices that are all reachable from each other

Cycle - a path (sequence of edges) that reaches the vertex it originated from

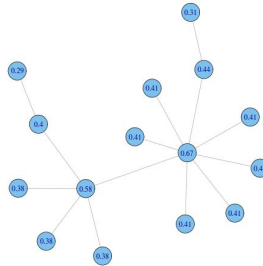


Graph Properties

Centrality - how “important” a vertex is. Can be computed in various ways (degree, betweenness, closeness)

Connectivity (cohesion) - how many vertices are reachable from other vertices

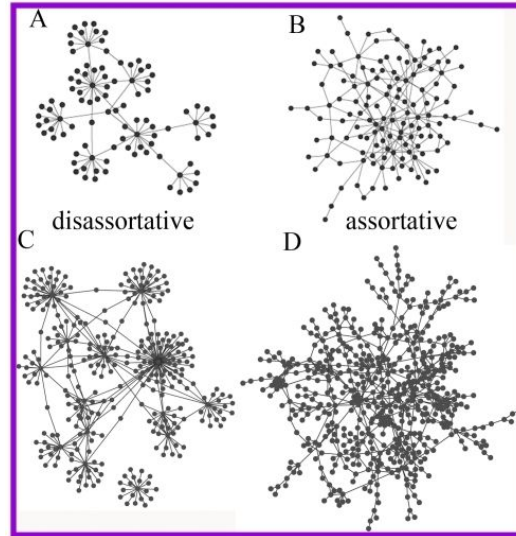
Diameter - max of how long it takes to get from one vertex to another



Graph Properties

Assortativity - similarity (in terms of degree) of pairs of linked vertices

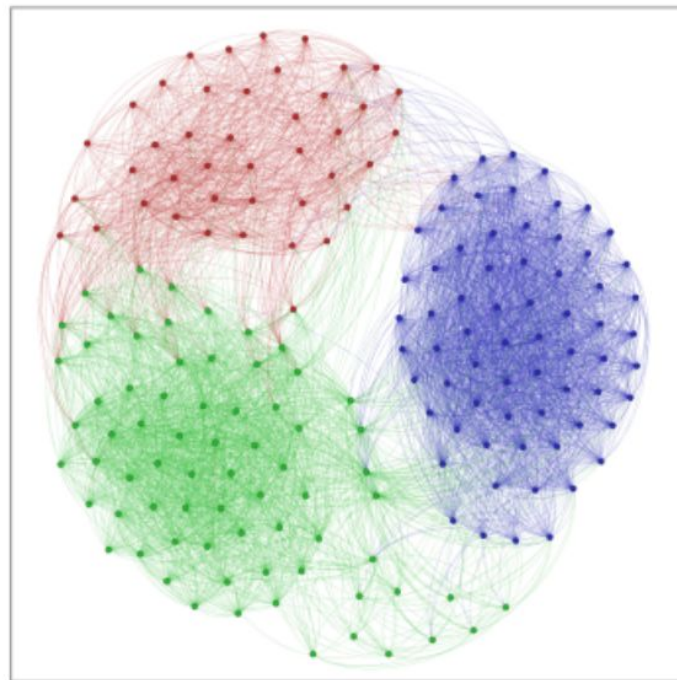
Rich-club coefficient - frequency with which nodes with high degree also connect to each other



Clustering on Graphs

We've seen k -means and hierarchical clustering.

Spectral clustering is another method of clustering, specifically for networked data.



Spectral Clustering

To perform spectral clustering on a graph [1]:

1. Generate the **Laplacian matrix** of the graph. $L = D - A$ (D has degree values on diagonal, A is the adjacency matrix).
2. Find the first k eigenvectors of this matrix.
3. Create a matrix where the columns are these eigenvectors. Let the rows be interpreted as data points in space.
4. Plug these points into some other clustering algorithm like k -means.



Coming Up

Your assignment: Project 2

Next week: Big data tools and wrap-up

See you then!

