OPENGOV DATA SCIENCE

Big Data NLP

Matt Seal

OPENGOV

Code shown with slide content:
https://github.com/MSeal/kdd2016

Bulk PDF Parsing

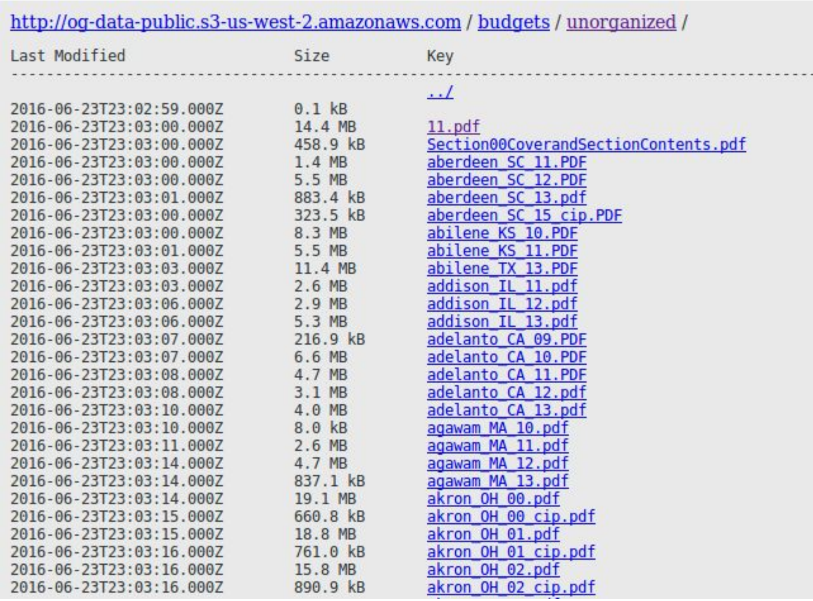
PDFs, and often individual pages of PDFS, are easily separable and independently processable. This means tokenizing the sentences or words in these documents is stupidly parallelizable and easy to distribute.

To handle this task one can either stream PDF files individually, or read them from a distributed store. The latter is easy to setup and easily accessible through Amazon S3.

S3 Bucket Viewing

```
<html>

<head></head>

<body>

  <div id="navigation"></div>

  <div id="listing"></div>

  <script
src="http://ajax.googleapis.com/ajax/libs/jquery/1.9.0/jquery.min.js">

  </script>

  <script type="text/javascript">

        var S3BL_IGNORE_PATH = true;

        var BUCKET_URL = 'http://og-data-public.s3-us-west-2.amazonaws.com';

  </script>

  <script src="http://rgrp.github.io/s3-bucket-listing/list.js"></script>

</body>

</html>
```

XML Parsed View



http://og-data-public.s3-us-west-2.amazonaws.com / budgets / unorganized /

```
Last Modified                Size          Key
------------------------------------------------------------------------------
                                           ../
2016-06-23T23:02:59.000Z     0.1 kB
2016-06-23T23:03:00.000Z     14.4 MB       11.pdf
2016-06-23T23:03:00.000Z     458.9 kB      Section00CoverandSectionContents.pdf
2016-06-23T23:03:00.000Z     1.4 MB        aberdeen_SC_11.PDF
2016-06-23T23:03:00.000Z     5.5 MB        aberdeen_SC_12.PDF
2016-06-23T23:03:01.000Z     883.4 kB      aberdeen_SC_13.pdf
2016-06-23T23:03:00.000Z     323.5 kB      aberdeen_SC_15_cip.PDF
2016-06-23T23:03:00.000Z     8.3 MB        abilene_KS_10.PDF
2016-06-23T23:03:01.000Z     5.5 MB        abilene_KS_11.PDF
2016-06-23T23:03:03.000Z     11.4 MB       abilene_TX_13.PDF
2016-06-23T23:03:03.000Z     2.6 MB        addison_IL_11.pdf
2016-06-23T23:03:06.000Z     2.9 MB        addison_IL_12.pdf
2016-06-23T23:03:06.000Z     5.3 MB        addison_IL_13.pdf
2016-06-23T23:03:07.000Z     216.9 kB      adelanto_CA_09.PDF
2016-06-23T23:03:07.000Z     6.6 MB        adelanto_CA_10.PDF
2016-06-23T23:03:08.000Z     4.7 MB        adelanto_CA_11.PDF
2016-06-23T23:03:08.000Z     3.1 MB        adelanto_CA_12.pdf
2016-06-23T23:03:10.000Z     4.0 MB        adelanto_CA_13.pdf
2016-06-23T23:03:10.000Z     8.0 kB        agawam_MA_10.pdf
2016-06-23T23:03:11.000Z     2.6 MB        agawam_MA_11.pdf
2016-06-23T23:03:14.000Z     4.7 MB        agawam_MA_12.pdf
2016-06-23T23:03:14.000Z     837.1 kB      agawam_MA_13.pdf
2016-06-23T23:03:14.000Z     19.1 MB       akron_OH_00.pdf
2016-06-23T23:03:15.000Z     660.8 kB      akron_OH_00_cip.pdf
2016-06-23T23:03:15.000Z     18.8 MB       akron_OH_01.pdf
2016-06-23T23:03:16.000Z     761.0 kB      akron_OH_01_cip.pdf
2016-06-23T23:03:16.000Z     15.8 MB       akron_OH_02.pdf
2016-06-23T23:03:16.000Z     890.9 kB      akron_OH_02_cip.pdf
```

Bulk PDF Parsing -- Sentence Tokens

- **Sentence one is here**
- **Sentence two is here**
- **This is a sentence broken apart**
- **Multiple sentences**
  - **This sentence is really long and probably includes multiple sentences**
  - **This can be a problem**
  - **<Noise>**
  - **Parsing is difficult here and common in pdfs**

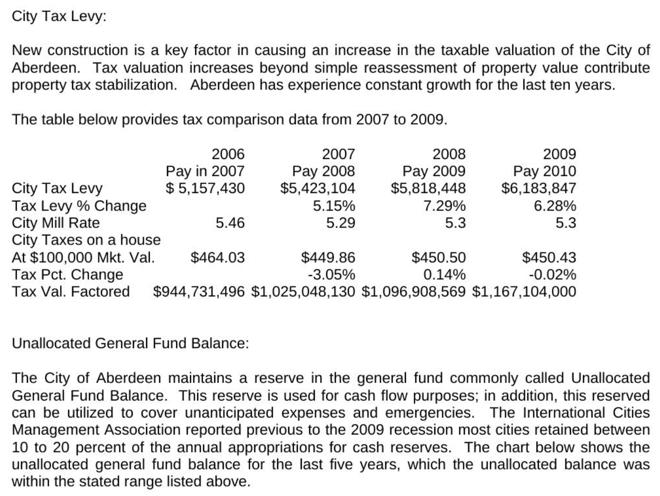
```
Sentence one is here. Sentence two

is here. This

sentence

is

broken apart.

This sentence is really long and
probably includes multiple sentences
this can be a problem parsing13more

thingsmashedt0gether parsing is
difficult here and common in pdfs
```

Bulk PDF Parsing -- Actual Text

Ideal Tokens

- **City tax Levy**
- **New construction is a key factor in causing an increase in the taxable valuation of the City of Aberdeen.**
- **...**
- **City Taxes on a house At $100,000 Mkt. Val.**
- **...**

Raw Text



City Tax Levy:

New construction is a key factor in causing an increase in the taxable valuation of the City of Aberdeen. Tax valuation increases beyond simple reassessment of property value contribute property tax stabilization. Aberdeen has experience constant growth for the last ten years.

The table below provides tax comparison data from 2007 to 2009.

	2006	2007	2008	2009
	Pay in 2007	Pay 2008	Pay 2009	Pay 2010
City Tax Levy	$ 5,157,430	$5,423,104	$5,818,448	$6,183,847
Tax Levy % Change		5.15%	7.29%	6.28%
City Mill Rate	5.46	5.29	5.3	5.3
City Taxes on a house				
At $100,000 Mkt. Val.	$464.03	$449.86	$450.50	$450.43
Tax Pct. Change		-3.05%	0.14%	-0.02%
Tax Val. Factored	$944,731,496	$1,025,048,130	$1,096,908,569	$1,167,104,000

Unallocated General Fund Balance:

The City of Aberdeen maintains a reserve in the general fund commonly called Unallocated General Fund Balance. This reserve is used for cash flow purposes; in addition, this reserved can be utilized to cover unanticipated expenses and emergencies. The International Cities Management Association reported previous to the 2009 recession most cities retained between 10 to 20 percent of the annual appropriations for cash reserves. The chart below shows the unallocated general fund balance for the last five years, which the unallocated balance was within the stated range listed above.

Bulk PDF Parsing -- Sentence Structures

It is common for papers to format text one way and maintain that format throughout. However, the structures can be represented in the PDF format in multiple ways.

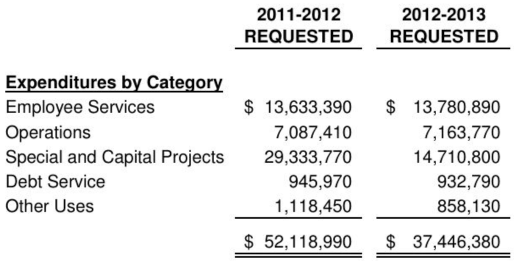
There can be one row of text with whitespace added to simulate two columns. It can be stored in a table object with invisible lines. And it can be a mix of text and vector maps

it is also possible for a PDF to change format in the middle of the document. What order do these sentences appear? Should we map the end of the top right box, with no punctuation, to the lowercased first word of the sentence at the top of this box?

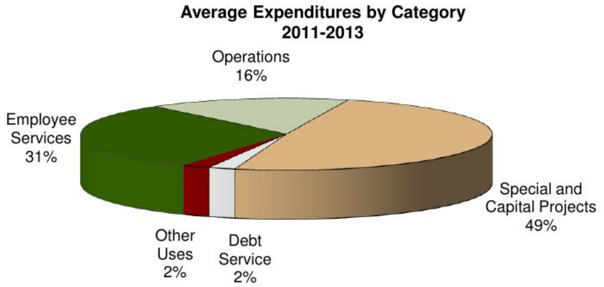
Bulk PDF Parsing -- Information Loss

There's an inherent accuracy to any given approach of PDF parsing. You will have some level of information loss in anything but well formated, pure text PDFs. **Tables in particular are difficult to parse** because they can be represented many different ways:

.. space separated, vectors, images, table objects, a mix of several approaches, etc...



	2011-2012 REQUESTED	2012-2013 REQUESTED
Expenditures by Category		
Employee Services	$ 13,633,390	$ 13,780,890
Operations	7,087,410	7,163,770
Special and Capital Projects	29,333,770	14,710,800
Debt Service	945,970	932,790
Other Uses	1,118,450	858,130
	$ 52,118,990	$ 37,446,380



Bulk PDF Parsing -- Noise Reduction

There are several tools to help reduce noise and information loss.

- Parsers
  - pdftotext
  - pdfminer (python)
  - PDFBox (java)
- OCR
  - Tesseract
  - Docsplit (ruby)
- Table extractor
  - Tabula
  - Pdftables (python)
- Text Repair
  - Hunspell (cyhunspell python)

Bulk PDF Parsing -- Outputs

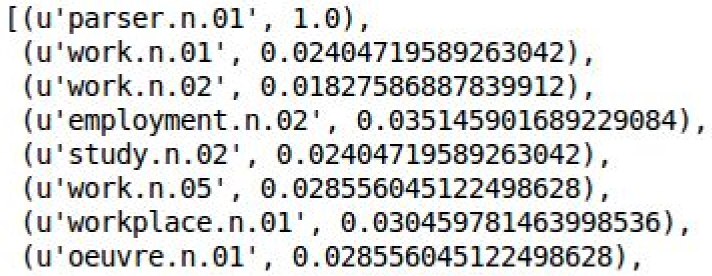
Once extraction and cleanup of PDF content is achieved at acceptable accuracies, there’s several useful artifacts that can be extracted for domain enrichment.

- Corpuses
    - Words
    - Senses
    - Lemmas
    - Information Content
- Search indexes
- Topics
- Named Entities

Bulk PDF Parsing -- Synset Corpus

Given text data in sentence structures, we can extract Synset Distributions which are translatable into Information Content.

$$IC(c) = -\log(P(c))$$



```
[(u'parser.n.01', 1.0),
 (u'work.n.01', 0.02404719589263042),
 (u'work.n.02', 0.01827586887839912),
 (u'employment.n.02', 0.03514590168922908 4),
 (u'study.n.02', 0.02404719589263042),
 (u'work.n.05', 0.02855604512249862 8),
 (u'workplace.n.01', 0.03045978146399853 6),
 (u'oeuvre.n.01', 0.02855604512249862 8),
```

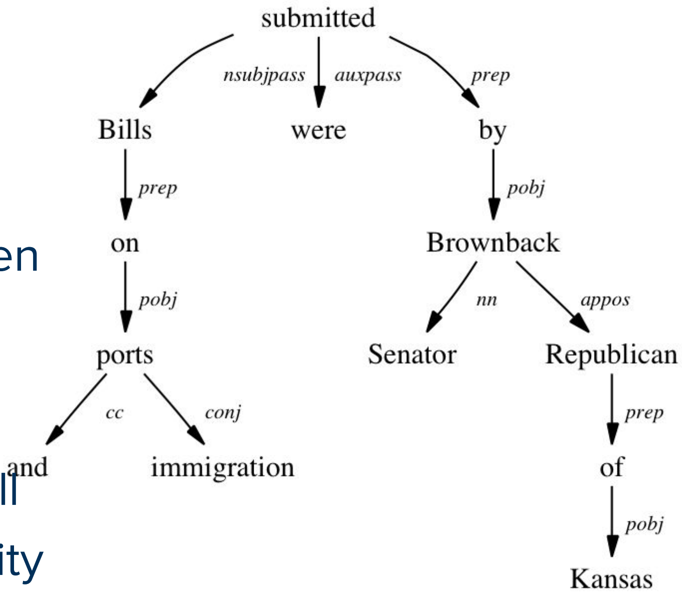
This is useful for many **semantic similarity** methods (Lin, Conrath, etc).

Synset Corpus Computation

Take sentences and find **similarity** between discovered senses and example sentences.

Word-overlap / vectorization is too sparse given number of samples.

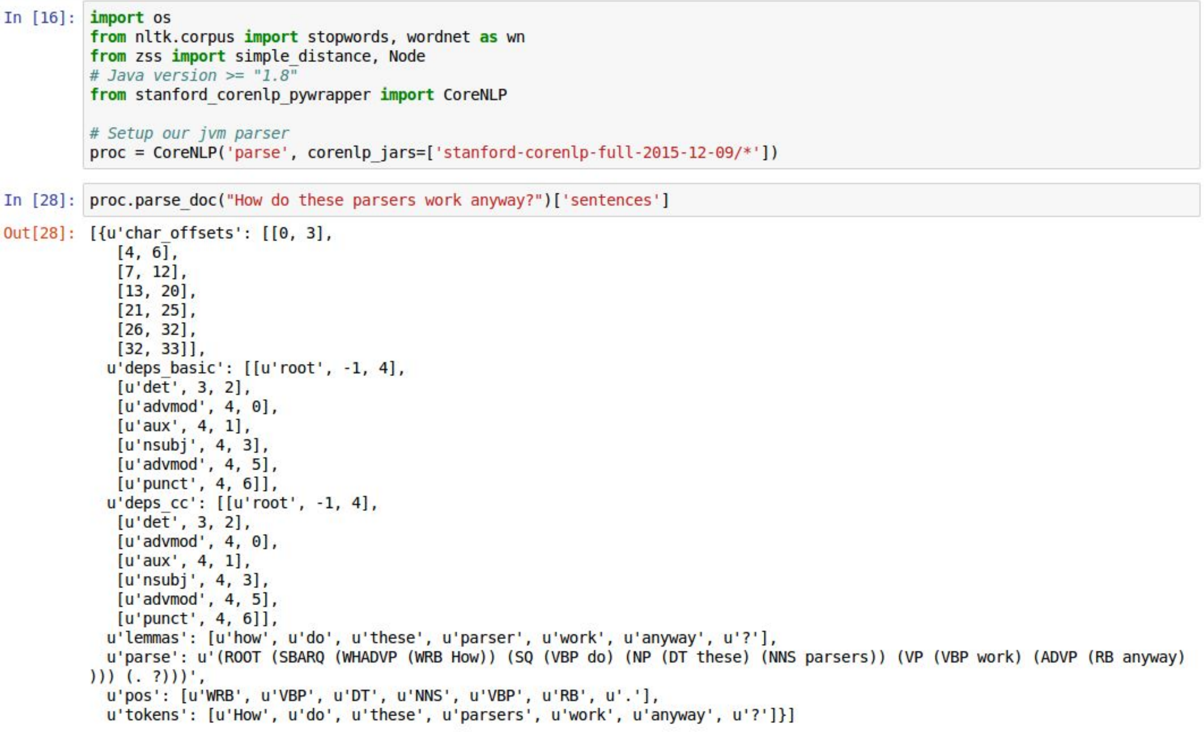
Expand to more **generic structure**: translate all sentences to parse trees and compute similarity of trees.



Source: http://nlp.stanford.edu/software/stanford-dependencies.shtml

Synset Corpus Computation Code

Let’s see some python code to achieve this computation and how we can distribute it!



```
In [16]: import os
         from nltk.corpus import stopwords, wordnet as wn
         from zss import simple_distance, Node
         # Java version >= "1.8"
         from stanford_corenlp_pywrapper import CoreNLP

         # Setup our jvm parser
         proc = CoreNLP('parse', corenlp_jars=['stanford-corenlp-full-2015-12-09/*'])

In [28]: proc.parse_doc("How do these parsers work anyway?")['sentences']

Out[28]: [{u'char_offsets': [[0, 3],
            [4, 6],
            [7, 12],
            [13, 20],
            [21, 25],
            [26, 32],
            [32, 33]],
           u'deps_basic': [[u'root', -1, 4],
            [u'det', 3, 2],
            [u'advmod', 4, 0],
            [u'aux', 4, 1],
            [u'nsubj', 4, 3],
            [u'advmod', 4, 5],
            [u'punct', 4, 6]],
           u'deps_cc': [[u'root', -1, 4],
            [u'det', 3, 2],
            [u'advmod', 4, 0],
            [u'aux', 4, 1],
            [u'nsubj', 4, 3],
            [u'advmod', 4, 5],
            [u'punct', 4, 6]],
           u'lemmas': [u'how', u'do', u'these', u'parser', u'work', u'anyway', u'?'],
           u'parse': u'(ROOT (SBARQ (WHADVP (WRB How)) (SQ (VBP do) (NP (DT these) (NNS parsers)) (VP (VBP work) (ADVP (RB anyway)
         ))) (. ?)))',
           u'pos': [u'WRB', u'VBP', u'DT', u'NNS', u'VBP', u'RB', u'.'],
           u'tokens': [u'How', u'do', u'these', u'parsers', u'work', u'anyway', u'?']}]
```

https://github.com/MSeal/kdd2016/blob/master/Synset%20Corpus.ipynb

Scaling Topic Clustering

Given text from new sources, what can we say about the data we're viewing?

Let's use some unsupervised techniques to find out.

How can we grow this into a distributed problem?

Split the raw data transformation work onto separate machines and reuse common distributed query patterns.

Unknown Number of Classes

When exploring **textual data** or **unstructured documents** the number of classes within that data isn't usually known.

Discovery and mapping of an ontology, or identifying social content trends for a market -- these tasks have variable numbers of classes and groupings.

Unsupervised Techniques

Utility Optimization

- Apply utility optimization rewards to exploratory agent
- Learn optimal paths or spaces

Clustering

- Associate data in state space, relative to other points
- Identify nearby groups in state space

Clustering of Sentences

There are many options for clustering:

- **Centroid** (e.g. k-means)
- **Distribution** (e.g. EM)
- **Hierarchical** (e.g. agglomerative)

Which arrive from two main flavors of approach:

- **Bottom-Up**
- **Top-Down**

Hierarchical Agglomerative Clustering

Newman Greedy method changes O(m*n^2) time to **O(m*n)**

Generates a Dendrogram of relationships

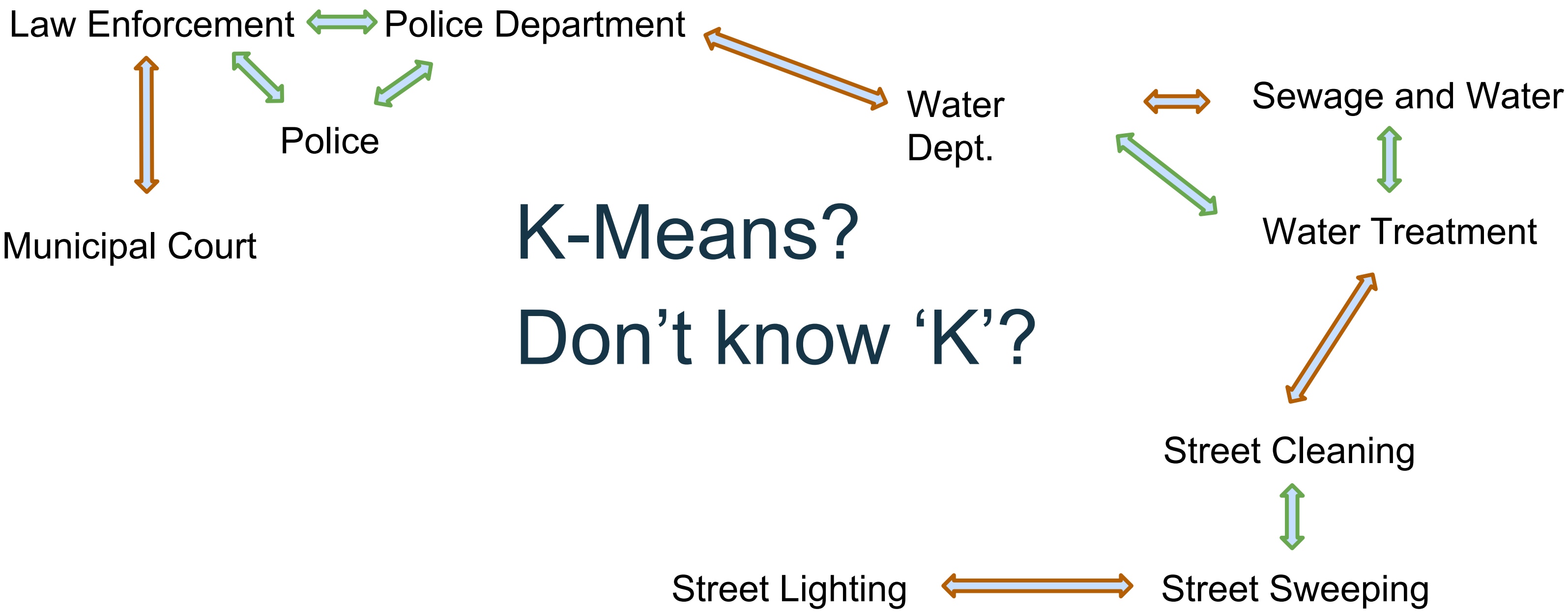
- Allows for slicing at any number of clusters
- Gives default option of local minima slice

Resistant to missing edges/data

No starting state required

No predefined number of classes required

Unsupervised Grouping



Law Enforcement

Police Department

Police

Municipal Court

Water Dept.

Sewage and Water

Water Treatment

K-Means?

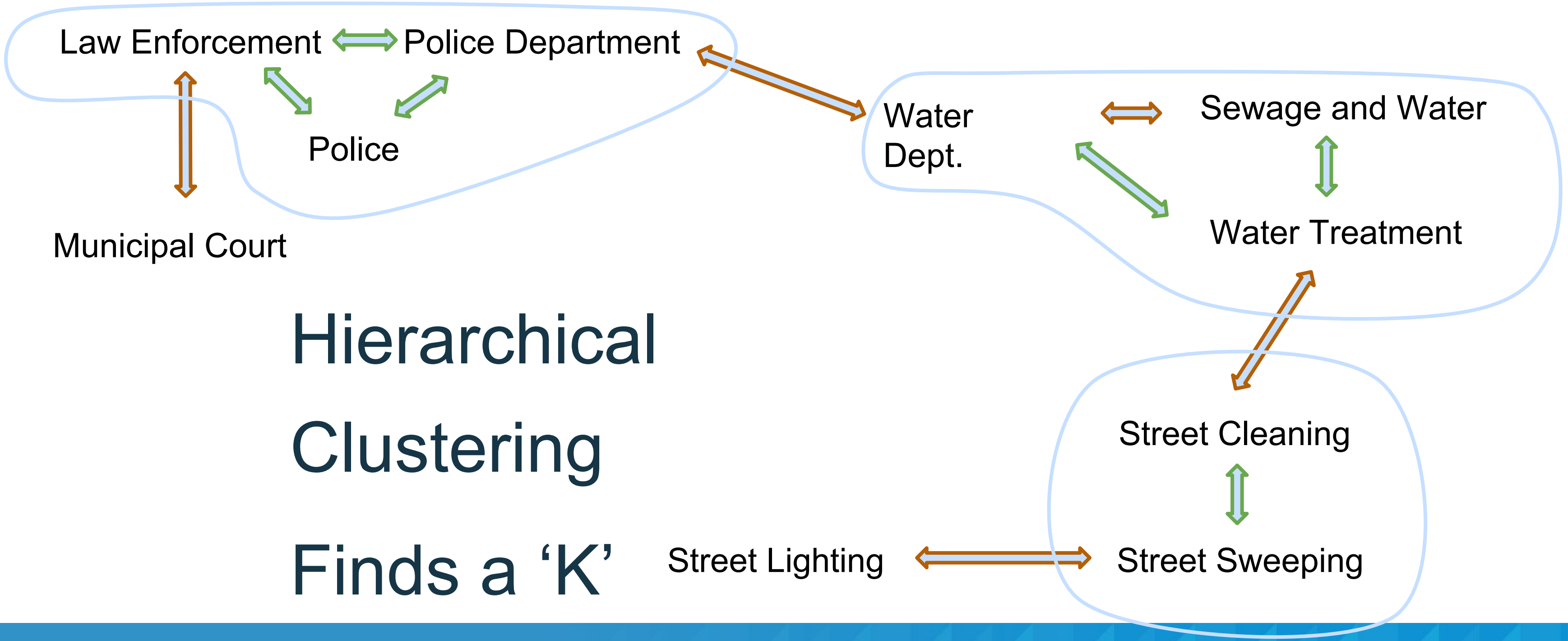
Don’t know ‘K’?

Street Cleaning

Street Lighting

Street Sweeping

Unsupervised Grouping Agglomerative



Hierarchical

Clustering

Finds a ‘K’

Agglomerative Dendrogram

Slicing the dendrogram at any number of clusters produces a valid arrangement of clusters.

The algorithm also **produces a score** for each number of clusters, of which you can select the maximum.

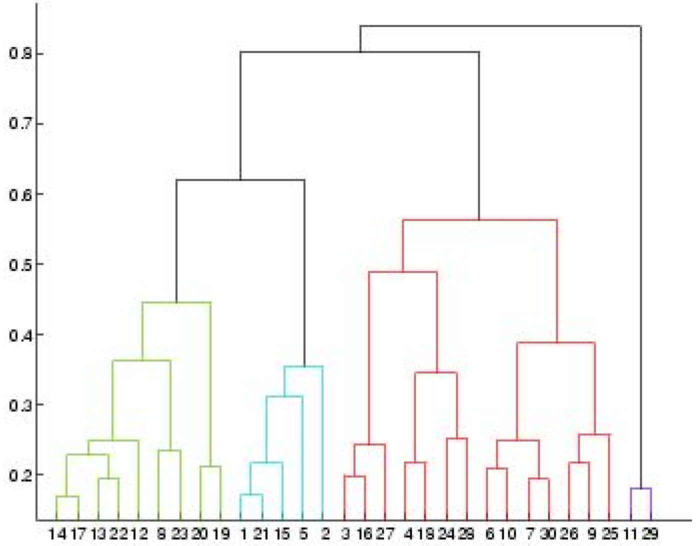
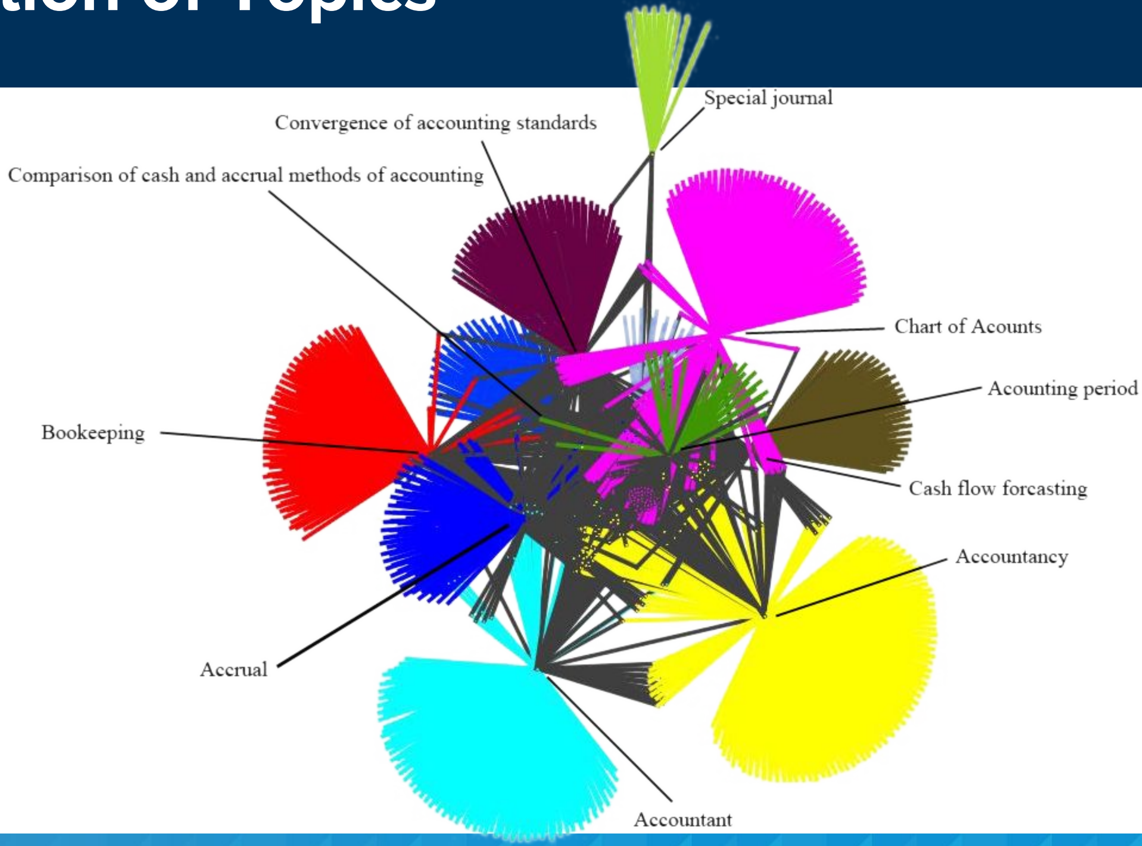


Image Source: http://www-rohan.sdsu.edu/doc/matlab/toolbox/stats/dendrogram.html

Clustering Visualization of Topics

This shows a visual representation of clusters pulled from Government Accounting sources.

Several classes of topics appeared from the raw text **by state space distance** associations.



https://github.com/MSeal/kdd2016/blob/master/Budget%20Topic%20Extraction.ipynb

Scaling Unsupervised Approaches

Most of the computation time for problems is in edge generation. These edges can be built in complete independence.

Eventually the clustering stage becomes the bottleneck. Then you can choose a distributed implementation of clustering from available options.

Experimentally, on AWS a single CPU can compute up to 20 million edges for 200k vertices in a under a few hours, 200 million edges for 2 million vertices in a week.