

# Topic Model Network Visualization

Lynn Cherny (@arnicas)  
lynn@ghostweather.com

August 2014

**WHAT IS TOPIC ANALYSIS?**

# Problems We're Attacking

- Document collections are hard work to explore/manage manually
- Sometimes the contents are completely or mostly “unknown” (e.g., an email archive, or a collection of research papers)
- We'd like at least semi-automated methods to group them, annotate them, explore relationships

# The Topic Problem

Text 1

We present a statistical parsing framework for sentence-level **sentiment** classification in this article. Different from previous work employing linguistic parsing results for **sentiment** analysis, we develop a statistical parser to directly analyze the **sentiment** structure of a sentence. We show that the complicated phenomena in **sentiment** analysis (e.g., negation, intensification, and contrast) can be elegantly handled the same as simple and straightforward

Text 2

**Sentiment** analysis of **Twitter** data is performed. The researcher has made the following contributions via this paper: (1) an innovative method for deriving **sentiment** score dictionaries using an existing **sentiment** dictionary as seed words is explored, and (2) an analysis of **clustered** tweet **sentiment** scores based on tweet length is performed.

Text 3

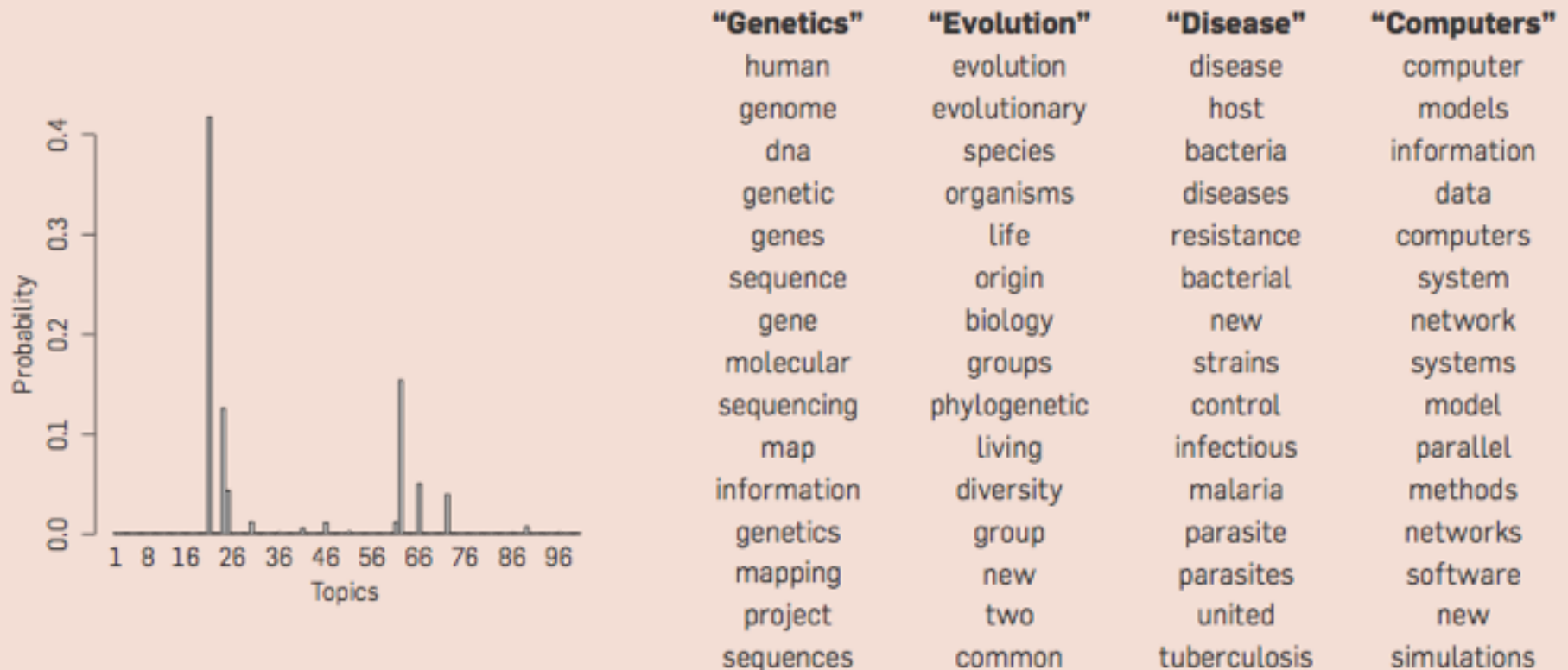
We perform a large-scale **linguistic** analysis of language diatopic variation using geotagged microblogging datasets. By collecting all **Twitter** messages written in Spanish over more than two years, we build a corpus from which a carefully selected list of concepts allows us to characterize Spanish varieties on a global scale. A **cluster** analysis proves the existence of well defined macroregions sharing common lexical properties.

# Intuitions

- Documents are composed of multiple words (“bag of words”). Documents may express multiple topics, using those words.
- Topics are considered unknown in advance — or “latent” — this is the problem we are trying to solve for.

## “LDA”: Latent Dirichlet Allocation

**Figure 2. Real inference with LDA.** We fit a 100-topic LDA model to 17,000 articles from the journal *Science*. At left are the inferred topic proportions for the example article in Figure 1. At right are the top 15 most frequent words from the most frequent topics found in this article.



## Topics

gene 0.04  
dna 0.02  
genetic 0.01  
...

life 0.02  
evolve 0.01  
organism 0.01  
...

brain 0.04  
neuron 0.02  
nerve 0.01  
...

data 0.02  
number 0.02  
computer 0.01  
...

## Documents

### Seeking Life's Bare (Genetic) Necessities

COLD SPRING HARBOR, NEW YORK—How many genes does an organism need to survive? Last week at the genome meeting here,\* two genome researchers with radically different approaches presented complementary views of the basic genes needed for life. One research team, using computer analyses to compare known genomes, concluded that today's organisms can be sustained with just 250 genes, and that the earliest life forms required a mere 128 genes. The other researcher mapped genes in a simple parasite and estimated that for this organism, 800 genes are plenty to do the job—but that anything short of 100 wouldn't be enough.

Although the numbers don't match precisely, those predictions

"are not all that far apart," especially in comparison to the 75,000 genes in the human genome, notes Siv Andersson, of Uppsala University in Sweden, who arrived at the 800 number. But coming up with a consensus answer may be more than just a genetic numbers game, particularly as more and more genomes are completely mapped and sequenced. "It may be a way of organizing any newly sequenced genome," explains Arcady Mushegian, a computational molecular biologist at the National Center for Biotechnology Information (NCBI) in Bethesda, Maryland. Comparing an

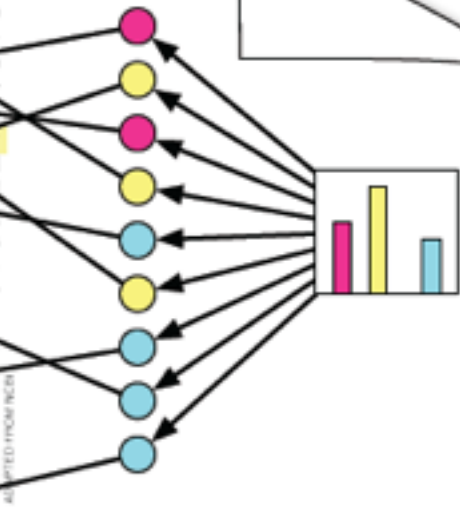


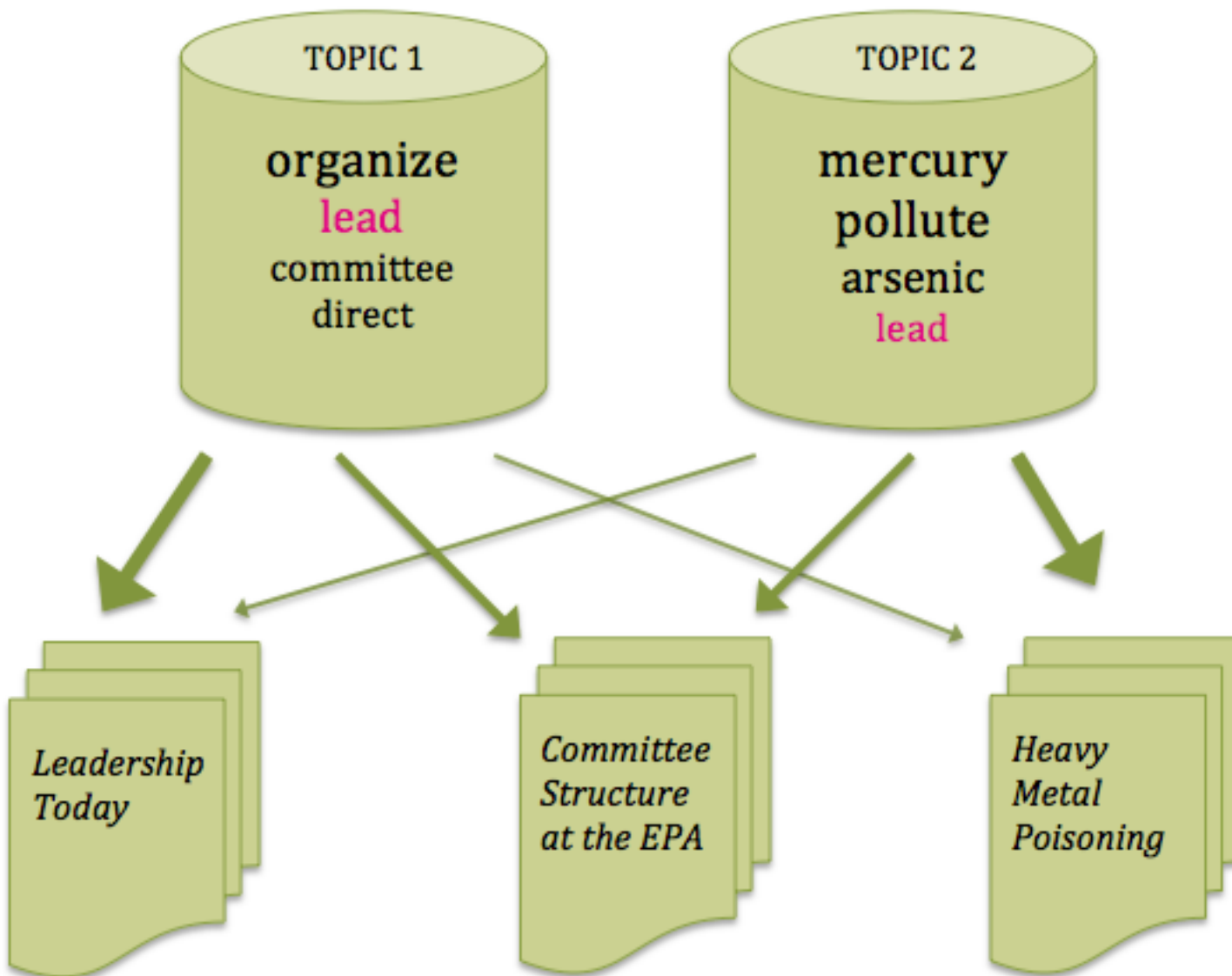
\* Genome Mapping and Sequencing, Cold Spring Harbor, New York, May 8 to 12.

**Stripping down.** Computer analysis yields an estimate of the minimum modern and ancient genomes.

SCIENCE • VOL. 272 • 24 MAY 1996

## Topic proportions and assignments







# **GETTING STARTED**

# Python Stuff you need

Python 2.7: ideally a virtual environment or install that includes libraries IPython notebook, nltk, pattern, pandas, numpy, networkx

1. Install miniconda: <http://conda.pydata.org/miniconda.html>.

2. Then:

```
>conda create -n topic_workshop ipython-notebook  
pandas numpy nltk pip
```

```
[accept the defaults]
```

```
>source activate topic_workshop
```

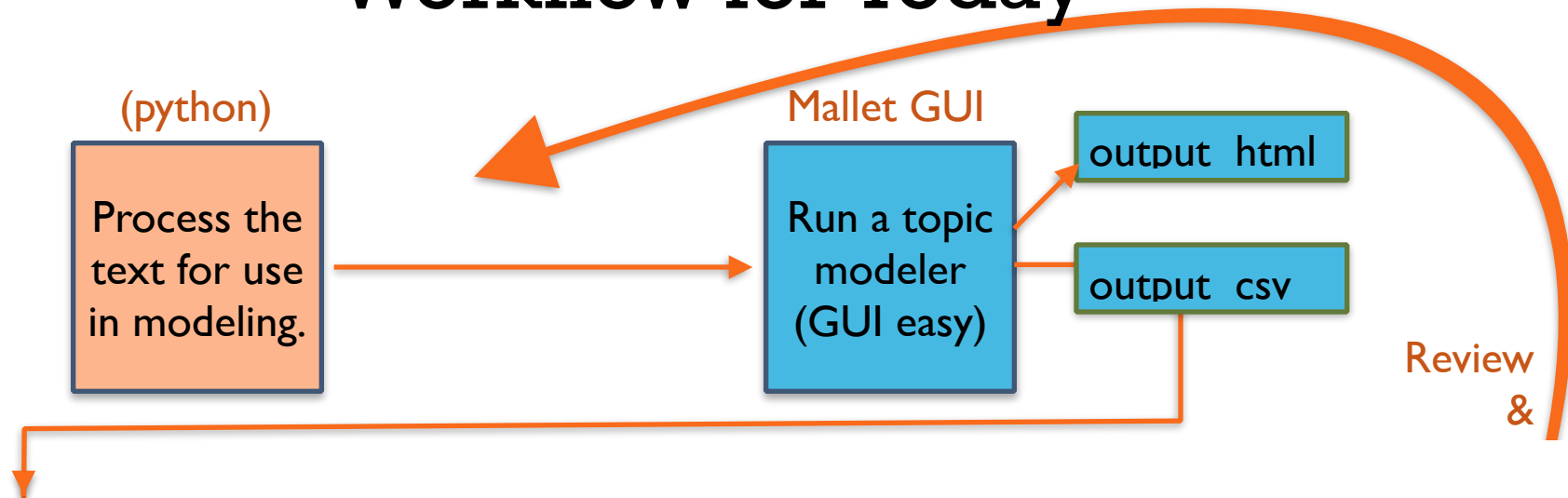
```
>pip install pattern
```

```
>pip install networkx
```

# Other Software

- TopicModelingTool.jar — provided
- Gephi ([gephi.org](http://gephi.org)) — download and install.
- Also add plugins for Sigma Exporter and Circular Layout.
- Note: If you have issues running gephi, read [this](#).

# Workflow for Today



# Cheat sheet: Scripts workflow

1. Optionally: Pre-process text files for part of speech...  
    `>python preprocess_files.py [original_dir] [new_dir] [part_of_speech]`
2. Run Mallet on text files.  
    Make an output directory. Double click on TopicModelingTool.jar.
3. Create gephi gdf files  
    `>python make_gephi_file.py [topics_dir_csv_path] [optl label]`
4. View in Gephi, fix layout....  
    Click on Gephi app and open your .gdf output file.
5. Optionally: Output gexf / json for d3  
    `>python d3_gexf.py [your output 'for excel' csvfile from 3] [label]`
6. From Gephi, export as sigma.js site  
    Find under export menu, if you installed that plugin
7. Fix up the sigma config and run server  
    `>python run_network.py [network_dir] [optl port]`

Make an directory for your output data (“my\_topics?”)

**RUN THE TOPIC MODELING  
TOOL.JAR**

# The Topic Modeling black box: David Newman's Topic Modeling Tool

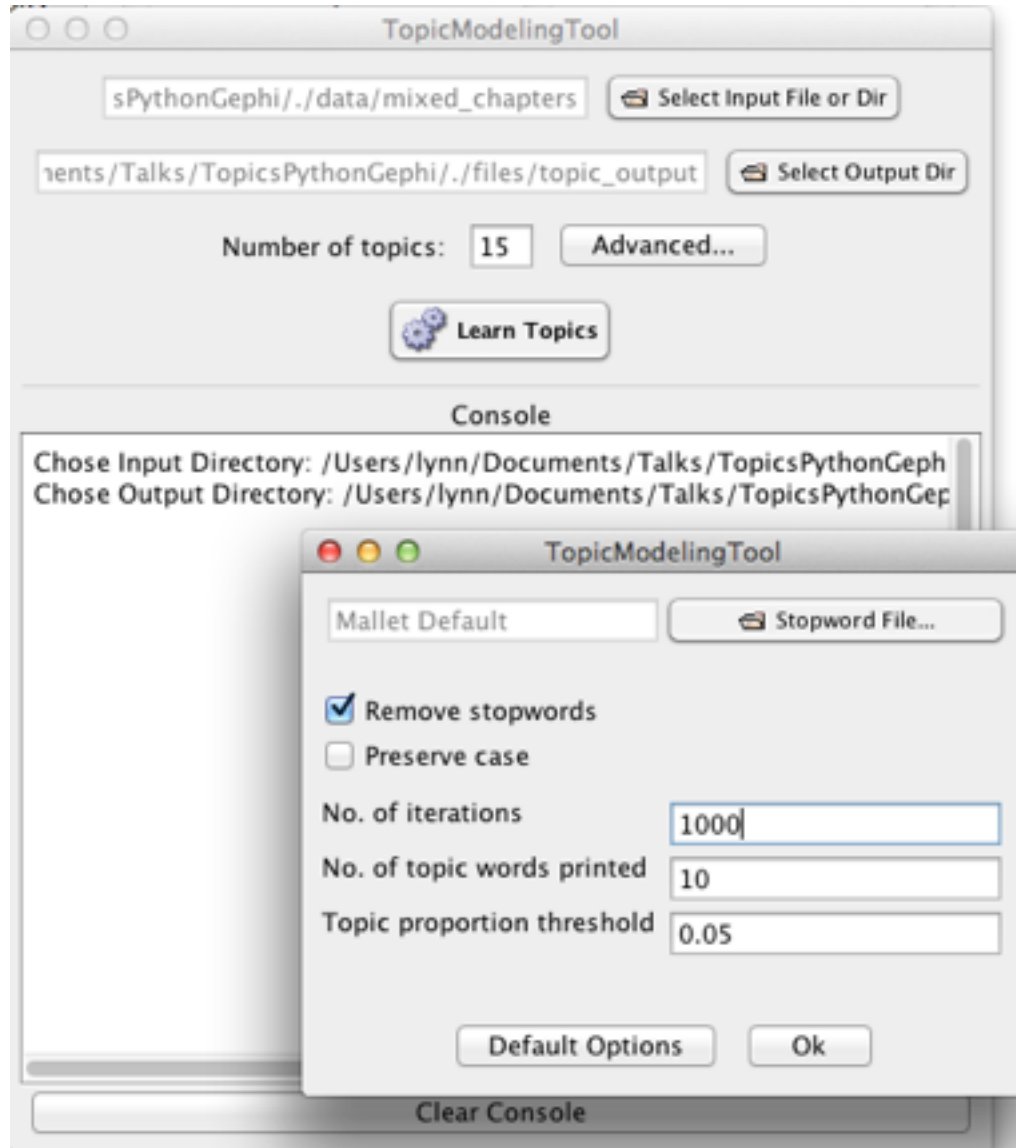
A tool available for non-technical audiences! A GUI wrapper on the state-of-the-art Mallet (a java-based app by David Mimno).

<https://code.google.com/p/topic-modeling-tool/>

(Also provided in the workshop files)

More of his work: <http://www.ics.uci.edu/~newman/>

# Topic Modeling Tool (GUI)



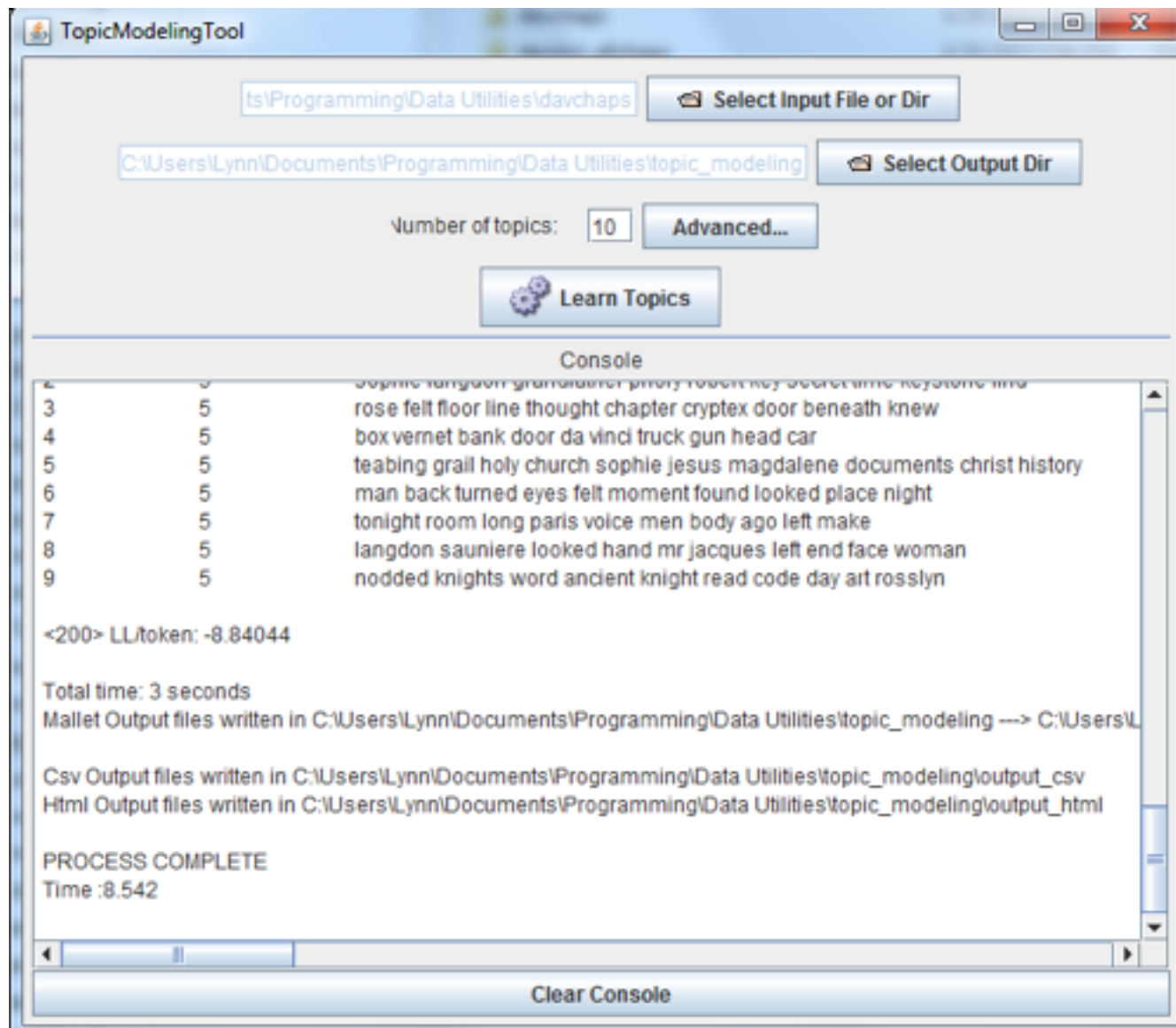
Create an output dir called "topic\_output" before you try to select.

Click "Advanced..."





# Post run...



# Understanding the Output


StackOverflow post: <http://stackoverflow.com/questions/8447393/how-to-understand-the-output-of-topic-model-class-in-mallet>

```
<1450> LL/token: -9.11846  
<1460> LL/token: -9.11803  
<1470> LL/token: -9.10896  
<1480> LL/token: -9.11237  
<1490> LL/token: -9.10845
```

Iteration number



Log Likelihood per word (we want this to increase as the algorithm runs)



# Output files: Data (csv), text web site

output\_csv

output\_html

List of T topics:

Topics\_Words.csv

List of topics in each of D  
documents:

TopicsInDocs.csv

List of top-ranked documents  
in each of T topics

DocsInTopics.csv

all\_topics.html

# Topic Modeling Mallet command line

You could also run mallet from the command line:

[http://programminghistorian.org/lessons/  
topic-modeling-and-mallet](http://programminghistorian.org/lessons/topic-modeling-and-mallet)

Or use a Python (or R) wrapper:

[http://radimrehurek.com/2014/03/tutorial-on-  
mallet-in-python/](http://radimrehurek.com/2014/03/tutorial-on-mallet-in-python/)

To do the rest of this workshop, you'd need to process the output files yourself similarly to our py code (assume \t seps, not csv)

# Pros/Cons vs CMD-Line Mallet

## Pros of GUI




- Allows stopword file input
- Takes folder or file of text
- Produces csv and html output in a neat dir structure
- Has a GUI! (simpler to just get going without code and help)
- A nice intro to using mallet on the command line

## Cons of GUI

- Runs with defaults, so no optimize-interval or other cmd line options
- No diagnostic output (a command-line option)
- Can get slightly fewer stats for your vis, as a result

# 2 of the 3 CSV Output files

	A	B	C	D	E	F	G	H
1	topicId	words..						
2		1	silas aringarosa remy teacher church dei opus bishop tomb vatican					
3		2	fache collet police message neveu agent phone captain plane sir					
4		3	sophie langdon grandfather priory robert key secret keystone time find					
5		4	rose floor felt line thought chapter cryptex door knew began					
6		5	box vernet bank vinci door head da louvre truck gun					
7		6	teabing grail holy church sophie jesus magdalene documents history ch					
8		7	man back turned felt eyes moment found looked place night					

 DocsInTopics.csv  
 Topics\_Words.csv  
 TopicsInDocs.csv

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
9		docid	filename	top topics	and contribution to doc ...														
10	2	1 C:\Users\I	9	0.21	4	0.188	8	0.147	7	0.13	5	0.13	6	0.063	1	0.055			
11	3	2 C:\Users\I	8	0.232	9	0.21	7	0.143	4	0.095	2	0.095	3	0.072	10	0.068	5	0.057	
	4	3 C:\Users\I	4	0.274	1	0.212	7	0.137	8	0.127	5	0.062	3	0.053					
	5	4 C:\Users\I	1	0.442	7	0.106	8	0.097	4	0.091	6	0.085							
	6	5 C:\Users\I	6	0.175	3	0.165	4	0.146	9	0.124	7	0.098	5	0.097	1	0.076	8	0.062	
	7	6 C:\Users\I	1	0.333	8	0.235	4	0.216	6	0.069									
	8	7 C:\Users\I	1	0.222	2	0.197	7	0.191	8	0.098	6	0.095	9	0.065					
	9	8 C:\Users\I	4	0.242	10	0.18	3	0.178	7	0.118	9	0.104	6	0.065	5	0.057			
	10	9 C:\Users\I	9	0.252	3	0.187	6	0.134	4	0.097	7	0.087	10	0.08	5	0.055	8	0.054	
	11	10 C:\Users\I	4	0.353	8	0.155	7	0.124	9	0.114	10	0.075	5	0.072					
	12	11 C:\Users\I	2	0.314	9	0.24	3	0.097	7	0.09	8	0.073	10	0.072					
	13	12 C:\Users\I	2	0.261	9	0.229	3	0.15	8	0.132	7	0.083							
	14	13 C:\Users\I	9	0.309	2	0.16	3	0.157	8	0.091	7	0.091	6	0.058					
	15	14 C:\Users\I	2	0.459	8	0.176	3	0.102	7	0.083	9	0.059							
	16	15 C:\Users\I	1	0.25	4	0.182	7	0.12	8	0.104	5	0.089	3	0.089	6	0.057			
	17	16 C:\Users\I	3	0.347	8	0.18	2	0.117	7	0.107	9	0.1	10	0.05	4	0.05			
	18	17 C:\Users\I	2	0.354	8	0.173	3	0.116	9	0.104	10	0.065	7	0.057	5	0.051			
	19	18 C:\Users\I	8	0.292	2	0.223	5	0.185	3	0.085	7	0.071	9	0.064	4	0.06			
	20	19 C:\Users\I	8	0.244	1	0.199	4	0.124	3	0.11	9	0.09	5	0.076	7	0.07	6	0.059	
	21	20 C:\Users\I	1	0.312	8	0.157	3	0.114	6	0.107	4	0.1	7	0.084	5	0.066			
	22	21 C:\Users\I	10	0.368	9	0.284	3	0.066	7	0.06	5	0.06	2	0.053					
	23	22 C:\Users\I	3	0.256	9	0.216	5	0.165	8	0.09	7	0.079	2	0.069	4	0.062			
	24	23 C:\Users\I	4	0.461	1	0.145	8	0.07	7	0.068	9	0.066	10	0.062					
	25	24 C:\Users\I	3	0.368	5	0.131	8	0.111	4	0.107	9	0.099	7	0.087					
	26	25 C:\Users\I	4	0.336	8	0.185	1	0.185	3	0.126	6	0.059							
	27	26 C:\Users\I	2	0.457	8	0.104	9	0.098	7	0.098	10	0.069	3	0.069	4	0.064			
	28	27 C:\Users\I	5	0.34	9	0.261	7	0.084	4	0.07	3	0.07	10	0.059					

This workshop has lots of code to process these files...  
(and a script: `make_gephi_file.py`)

```
In [36]: topics_per_doc = read_doctopics(topic_docs) # keep in mind the input GUI said to cut off classification a

chap_0 {'1': '0.055', '5': '0.130', '4': '0.188', '7': '0.130', '6': '0.063', '9': '0.210', '8': '0.147'}
chap_1 {'10': '0.068', '3': '0.072', '2': '0.095', '5': '0.057', '4': '0.095', '7': '0.143', '9': '0.210'}
chap_10 {'1': '0.212', '3': '0.053', '5': '0.062', '4': '0.274', '7': '0.137', '8': '0.127'}
chap_100 {'1': '0.442', '8': '0.097', '4': '0.091', '7': '0.106', '6': '0.085'}
chap_101 {'1': '0.076', '3': '0.165', '5': '0.097', '4': '0.146', '7': '0.098', '6': '0.175', '9': '0.124'}
chap_102 {'1': '0.333', '8': '0.235', '4': '0.216', '6': '0.069'}
chap_103 {'1': '0.222', '2': '0.197', '7': '0.191', '6': '0.095', '9': '0.065', '8': '0.098'}
chap_104 {'10': '0.180', '3': '0.178', '5': '0.057', '4': '0.242', '7': '0.118', '6': '0.065', '9': '0.10'}
chap_105 {'10': '0.080', '3': '0.187', '5': '0.055', '4': '0.097', '7': '0.087', '6': '0.134', '9': '0.25'}
chap_106 {'10': '0.075', '5': '0.072', '4': '0.353', '7': '0.124', '9': '0.114', '8': '0.155'}
chap_11 {'10': '0.072', '3': '0.097', '2': '0.314', '7': '0.090', '9': '0.240', '8': '0.073'}
chap_12 {'9': '0.229', '8': '0.132', '3': '0.150', '2': '0.261', '7': '0.083'}
chap_13 {'3': '0.157', '2': '0.160', '7': '0.091', '6': '0.058', '9': '0.309', '8': '0.091'}
chap_14 {'9': '0.059', '8': '0.176', '3': '0.102', '2': '0.459', '7': '0.083'}
```

# The GUI's HTML output is a little lacking...

**TOPIC : man back turned felt eyes moment found looked place night ...**

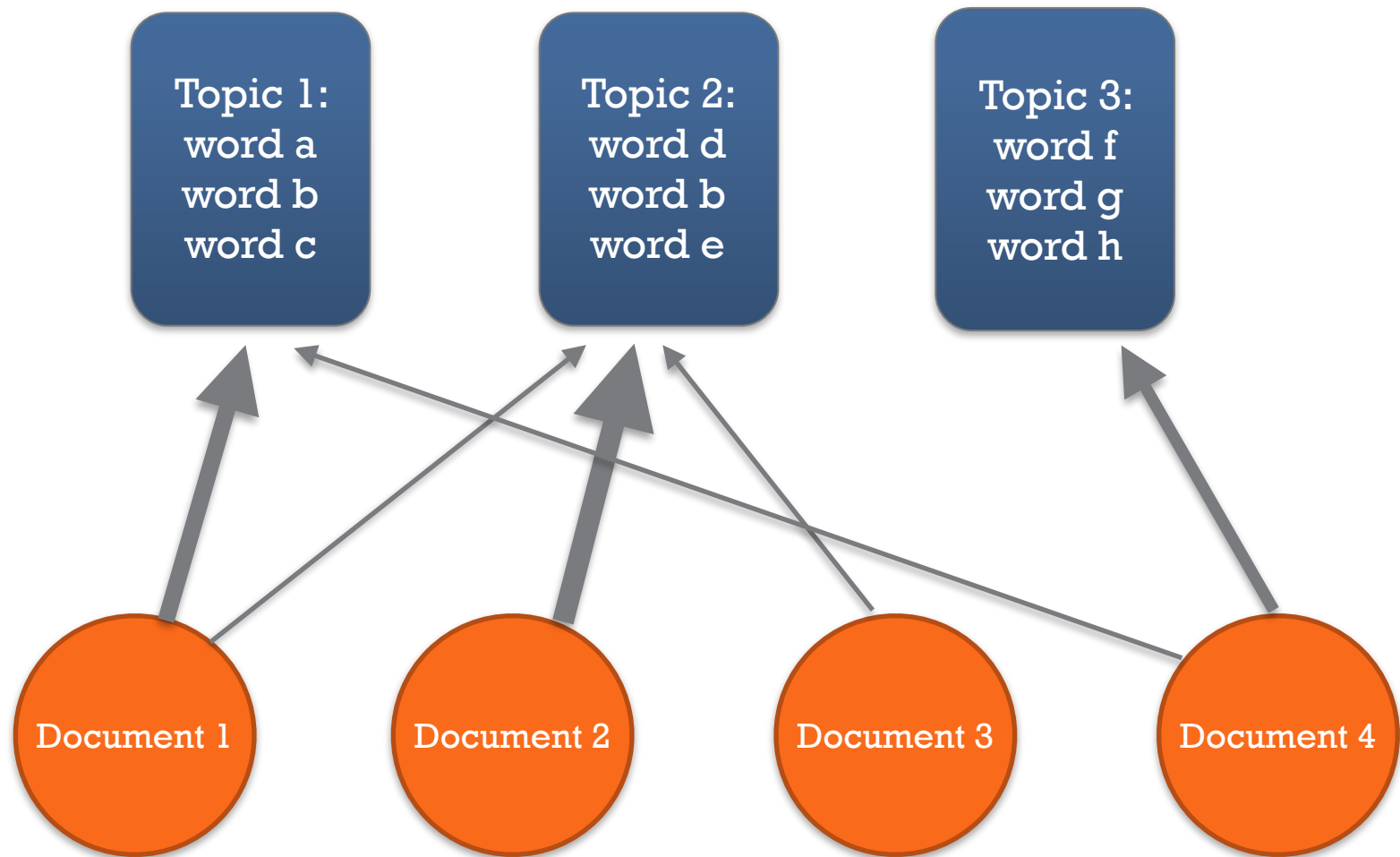
top-ranked docs in this topic (#words in doc assigned to this topic)

2. (219) chap\_67.txt
3. (193) chap\_104.txt
4. (180) chap\_84.txt
5. (179) chap\_99.txt
6. (160) chap\_51.txt
7. (153) chap\_32.txt
8. (145) chap\_81.txt

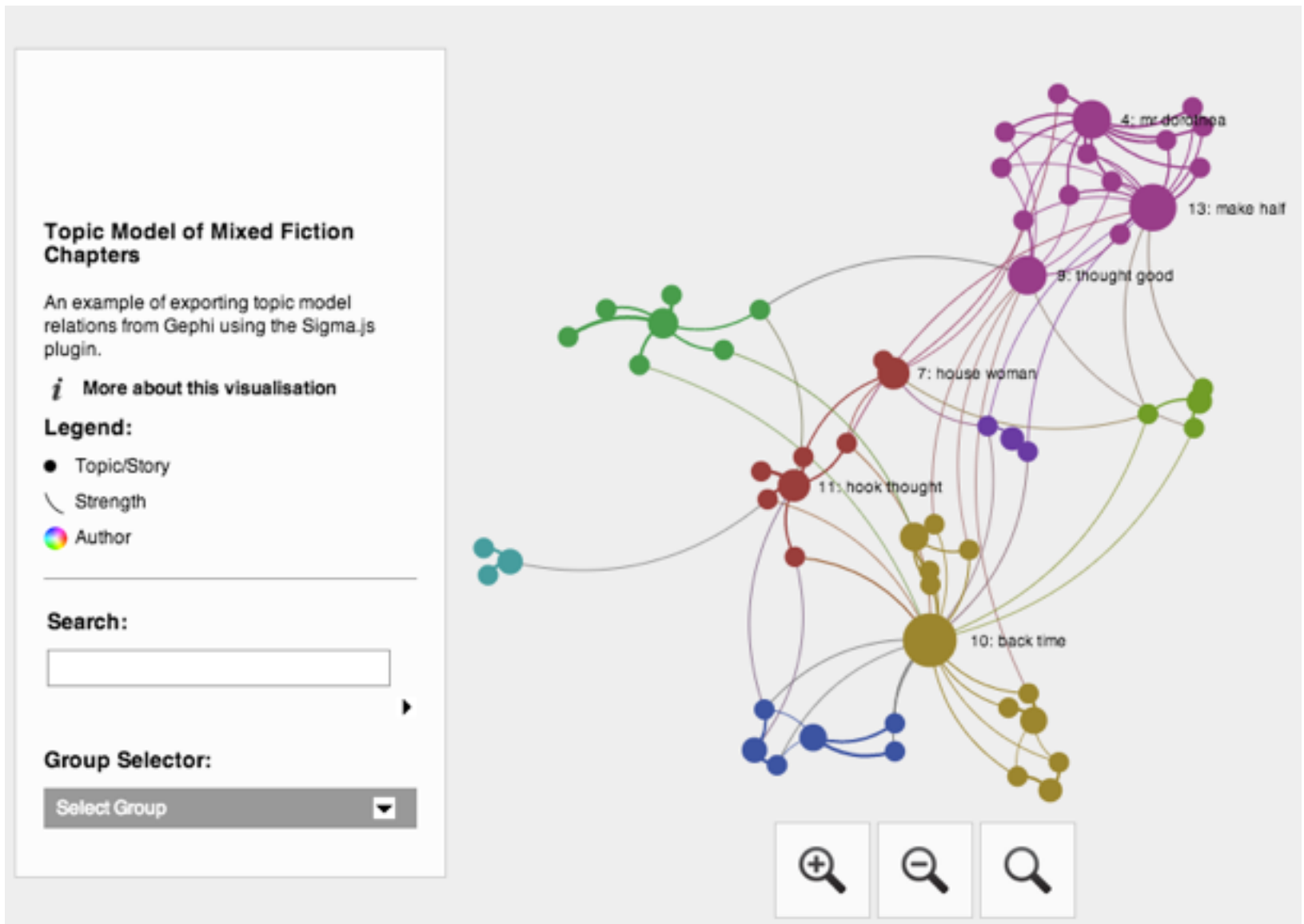
A bipartite graph of chapters and topics is an obvious vis method....



# The results of topic modeling



# Our Goal



# Our Goal

## Topic Model of Mixed Fiction Chapters

An example of exporting topic model relations from Gephi using the Sigma.js plugin.

 More about this visualisation

### Legend:

- Topic/Story
- Strength
- Author

### Search:

### Group Selector:

Select Group

lewis\_lionwitch\_ch12.txt

10: back time

 Return to the full network

### Information Pane

**10: back time**

author: Topic

Modularity Class: 1

text: Top words:back time face hand room head eyes ll door don

type: Topic

### Connections:

[barrie\\_peterpan\\_ch15.txt](#)  
[brown\\_davinci\\_ch1.txt](#)  
[conrad\\_secret\\_ch1.txt](#)  
[conrad\\_secret\\_ch2.txt](#)  
[doyle\\_redheadedleague.txt](#)  
[doyle\\_scandalbohemia.txt](#)  
[james\\_fiftyshades\\_ch1.txt](#)  
[james\\_fiftyshades\\_ch18.txt](#)  
[james\\_fiftyshades\\_ch2.txt](#)  
[james\\_fiftyshades\\_ch23.txt](#)  
[lewis\\_lionwitch\\_ch1.txt](#)  
[lewis\\_lionwitch\\_ch12.txt](#)  
[lewis\\_lionwitch\\_ch14.txt](#)  
[lewis\\_lionwitch\\_ch2.txt](#)  
[meyer\\_twilight\\_ch1.txt](#)  
[meyer\\_twilight\\_ch15.txt](#)  
[meyer\\_twilight\\_ch2.txt](#)  
[meyer\\_twilight\\_ch21.txt](#)  
[stevenson\\_treasure\\_ch1.txt](#)  
[stevenson\\_treasure\\_ch2.txt](#)

# **PROCESS CSV OUTPUT FROM GUI TOOL**

# Cheat sheet: Scripts workflow

1. Optionally: Pre-process text files for part of speech...  
    `>python preprocess_files.py [original_dir] [new_dir] [part_of_speech]`
2. Run Mallet on text files.  
    Make an output directory. Double click on TopicModelingTool.jar.
3. Process output: Create gephi gdf files  
    `>python make_gephi_file.py [topics_dir_csv_path] [optl label]`
4. View in Gephi, fix layout....  
    Click on Gephi app and open your .gdf output file.
5. Optionally: Output gexf / json for d3  
    `>python d3_gexf.py [your output 'for excel' csvfile from 3] [label]`
6. From Gephi, export as sigma.js site  
    Find under export menu, if you installed that plugin
7. Fix up the sigma config and run server  
    `>python run_network.py [network_dir] [optl port]`

# Our next step: Process GUI CSV Output

After running the Topic Modeling Tool, we start with the IPython notebook “Topic Analysis of Mixed Fiction.ipynb.”

If you want to run the notebooks, make sure you are in an active virtual environment, set up as I described.

```
> source activate topic_workshop
```

```
(topic_workshop)> ipython notebook
```

If you don't want to run it, you can achieve the same outputs with the path to the gui output csv file:

```
(topic_workshop)> cd files
```

```
(topic_workshop)> python make_gephi_file.py topic_output/output_csv all
```

# Our next steps... visualize

1. Excel pivot table analysis with the for\_excel.csv file
2. Gephi for the gdf file output!

Tips on Gephi layout and UI are in the PDF:

# GephiToSigmaJS\_Mixed.pdf

Item Name	Topic1	Topic2	Topic3	Topic4	Topic5	Topic6	Topic7	Topic8	Topic9	Topic10	Topic11	Topic12	Topic13	Topic14	Topic15	Topic16	Topic17	Topic18	Topic19	Topic20	Topic21	Topic22	Topic23	Topic24	Topic25	Topic26	Topic27	Topic28	Topic29	Topic30	Topic31	Topic32	Topic33	Topic34	Topic35	Topic36	Topic37	Topic38	Topic39	Topic40	Topic41	Topic42	Topic43	Topic44	Topic45	Topic46	Topic47	Topic48	Topic49	Topic50	Topic51	Topic52	Topic53	Topic54	Topic55	Topic56	Topic57	Topic58	Topic59	Topic60	Topic61	Topic62	Topic63	Topic64	Topic65	Topic66	Topic67	Topic68	Topic69	Topic70	Topic71	Topic72	Topic73	Topic74	Topic75	Topic76	Topic77	Topic78	Topic79	Topic80	Topic81	Topic82	Topic83	Topic84	Topic85	Topic86	Topic87	Topic88	Topic89	Topic90	Topic91	Topic92	Topic93	Topic94	Topic95	Topic96	Topic97	Topic98	Topic99	Topic100	Topic101	Topic102	Topic103	Topic104	Topic105	Topic106	Topic107	Topic108	Topic109	Topic110	Topic111	Topic112	Topic113	Topic114	Topic115	Topic116	Topic117	Topic118	Topic119	Topic120	Topic121	Topic122	Topic123	Topic124	Topic125	Topic126	Topic127	Topic128	Topic129	Topic130	Topic131	Topic132	Topic133	Topic134	Topic135	Topic136	Topic137	Topic138	Topic139	Topic140	Topic141	Topic142	Topic143	Topic144	Topic145	Topic146	Topic147	Topic148	Topic149	Topic150	Topic151	Topic152	Topic153	Topic154	Topic155	Topic156	Topic157	Topic158	Topic159	Topic160	Topic161	Topic162	Topic163	Topic164	Topic165	Topic166	Topic167	Topic168	Topic169	Topic170	Topic171	Topic172	Topic173	Topic174	Topic175	Topic176	Topic177	Topic178	Topic179	Topic180	Topic181	Topic182	Topic183	Topic184	Topic185	Topic186	Topic187	Topic188	Topic189	Topic190	Topic191	Topic192	Topic193	Topic194	Topic195	Topic196	Topic197	Topic198	Topic199	Topic200	Topic201	Topic202	Topic203	Topic204	Topic205	Topic206	Topic207	Topic208	Topic209	Topic210	Topic211	Topic212	Topic213	Topic214	Topic215	Topic216	Topic217	Topic218	Topic219	Topic220	Topic221	Topic222	Topic223	Topic224	Topic225	Topic226	Topic227	Topic228	Topic229	Topic230	Topic231	Topic232	Topic233	Topic234	Topic235	Topic236	Topic237	Topic238	Topic239	Topic240	Topic241	Topic242	Topic243	Topic244	Topic245	Topic246	Topic247	Topic248	Topic249	Topic250	Topic251	Topic252	Topic253	Topic254	Topic255	Topic256	Topic257	Topic258	Topic259	Topic260	Topic261	Topic262	Topic263	Topic264	Topic265	Topic266	Topic267	Topic268	Topic269	Topic270	Topic271	Topic272	Topic273	Topic274	Topic275	Topic276	Topic277	Topic278	Topic279	Topic280	Topic281	Topic282	Topic283	Topic284	Topic285	Topic286	Topic287	Topic288	Topic289	Topic290	Topic291	Topic292	Topic293	Topic294	Topic295	Topic296	Topic297	Topic298	Topic299	Topic300	Topic301	Topic302	Topic303	Topic304	Topic305	Topic306	Topic307	Topic308	Topic309	Topic310	Topic311	Topic312	Topic313	Topic314	Topic315	Topic316	Topic317	Topic318	Topic319	Topic320	Topic321	Topic322	Topic323	Topic324	Topic325	Topic326	Topic327	Topic328	Topic329	Topic330	Topic331	Topic332	Topic333	Topic334	Topic335	Topic336	Topic337	Topic338	Topic339	Topic340	Topic341	Topic342	Topic343	Topic344	Topic345	Topic346	Topic347	Topic348	Topic349	Topic350	Topic351	Topic352	Topic353	Topic354	Topic355	Topic356	Topic357	Topic358	Topic359	Topic360	Topic361	Topic362	Topic363	Topic364	Topic365	Topic366	Topic367	Topic368	Topic369	Topic370	Topic371	Topic372	Topic373	Topic374	Topic375	Topic376	Topic377	Topic378	Topic379	Topic380	Topic381	Topic382	Topic383	Topic384	Topic385	Topic386	Topic387	Topic388	Topic389	Topic390	Topic391	Topic392	Topic393	Topic394	Topic395	Topic396	Topic397	Topic398	Topic399	Topic400	Topic401	Topic402	Topic403	Topic404	Topic405	Topic406	Topic407	Topic408	Topic409	Topic410	Topic411	Topic412	Topic413	Topic414	Topic415	Topic416	Topic417	Topic418	Topic419	Topic420	Topic421	Topic422	Topic423	Topic424	Topic425	Topic426	Topic427	Topic428	Topic429	Topic430	Topic431	Topic432	Topic433	Topic434	Topic435	Topic436	Topic437	Topic438	Topic439	Topic440	Topic441	Topic442	Topic443	Topic444	Topic445	Topic446	Topic447	Topic448	Topic449	Topic450	Topic451	Topic452	Topic453	Topic454	Topic455	Topic456	Topic457	Topic458	Topic459	Topic460	Topic461	Topic462	Topic463	Topic464	Topic465	Topic466	Topic467	Topic468	Topic469	Topic470	Topic471	Topic472	Topic473	Topic474	Topic475	Topic476	Topic477	Topic478	Topic479	Topic480	Topic481	Topic482	Topic483	Topic484	Topic485	Topic486	Topic487	Topic488	Topic489	Topic490	Topic491	Topic492	Topic493	Topic494	Topic495	Topic496	Topic497	Topic498	Topic499	Topic500	Topic501	Topic502	Topic503	Topic504	Topic505	Topic506	Topic507	Topic508	Topic509	Topic510	Topic511	Topic512	Topic513	Topic514	Topic515	Topic516	Topic517	Topic518	Topic519	Topic520	Topic521	Topic522	Topic523	Topic524	Topic525	Topic526	Topic527	Topic528	Topic529	Topic530	Topic531	Topic532	Topic533	Topic534	Topic535	Topic536	Topic537	Topic538	Topic539	Topic540	Topic541	Topic542	Topic543	Topic544	Topic545	Topic546	Topic547	Topic548	Topic549	Topic550	Topic551	Topic552	Topic553	Topic554	Topic555	Topic556	Topic557	Topic558	Topic559	Topic560	Topic561	Topic562	Topic563	Topic564	Topic565	Topic566	Topic567	Topic568	Topic569	Topic570	Topic571	Topic572	Topic573	Topic574	Topic575	Topic576	Topic577	Topic578	Topic579	Topic580	Topic581	Topic582	Topic583	Topic584	Topic585	Topic586	Topic587	Topic588	Topic589	Topic590	Topic591	Topic592	Topic593	Topic594	Topic595	Topic596	Topic597	Topic598	Topic599	Topic600	Topic601	Topic602	Topic603	Topic604	Topic605	Topic606	Topic607	Topic608	Topic609	Topic610	Topic611	Topic612	Topic613	Topic614	Topic615	Topic616	Topic617	Topic618	Topic619	Topic620	Topic621	Topic622	Topic623	Topic624	Topic625	Topic626	Topic627	Topic628	Topic629	Topic630	Topic631	Topic632	Topic633	Topic634	Topic635	Topic636	Topic637	Topic638	Topic639	Topic640	Topic641	Topic642	Topic643	Topic644	Topic645	Topic646	Topic647	Topic648	Topic649	Topic650	Topic651	Topic652	Topic653	Topic654	Topic655	Topic656	Topic657	Topic658	Topic659	Topic660	Topic661	Topic662	Topic663	Topic664	Topic665	Topic666	Topic667	Topic668	Topic669	Topic670	Topic671	Topic672	Topic673	Topic674	Topic675	Topic676	Topic677	Topic678	Topic679	Topic680	Topic681	Topic682	Topic683	Topic684	Topic685	Topic686	Topic687	Topic688	Topic689	Topic690	Topic691	Topic692	Topic693	Topic694	Topic695	Topic696	Topic697	Topic698	Topic699	Topic700	Topic701	Topic702	Topic703	Topic704	Topic705	Topic706	Topic707	Topic708	Topic709	Topic710	Topic711	Topic712	Topic713	Topic714	Topic715	Topic716	Topic717	Topic718	Topic719	Topic720	Topic721	Topic722	Topic723	Topic724	Topic725	Topic726	Topic727	Topic728	Topic729	Topic730	Topic731	Topic732	Topic733	Topic734	Topic735	Topic736	Topic737	Topic738	Topic739	Topic740	Topic741	Topic742	Topic743	Topic744	Topic745	Topic746	Topic747	Topic748	Topic749	Topic750	Topic751	Topic752	Topic753	Topic754	Topic755	Topic756	Topic757	Topic758	Topic759	Topic760	Topic761	Topic762	Topic763	Topic764	Topic765	Topic766	Topic767	Topic768	Topic769	Topic770	Topic771	Topic772	Topic773	Topic774	Topic775	Topic776	Topic777	Topic778	Topic779	Topic780	Topic781	Topic782	Topic783	Topic784	Topic785	Topic786	Topic787	Topic788	Topic789	Topic790	Topic791	Topic792	Topic793	Topic794	Topic795	Topic796	Topic797	Topic798	Topic799	Topic800	Topic801	Topic802	Topic803	Topic804	Topic805	Topic806	Topic807	Topic808	Topic809	Topic810	Topic811	Topic812	Topic813	Topic814	Topic815	Topic816	Topic817	Topic818	Topic819	Topic820	Topic821	Topic822	Topic823	Topic824	Topic825	Topic826	Topic827	Topic828	Topic829	Topic830	Topic831	Topic832	Topic833	Topic834	Topic835	Topic836	Topic837	Topic838	Topic839	Topic840	Topic841	Topic842	Topic843	Topic844	Topic845	Topic846	Topic847	Topic848	Topic849	Topic850	Topic851	Topic852	Topic853	Topic854	Topic855	Topic856	Topic857	Topic858	Topic859	Topic860	Topic861	Topic862	Topic863	Topic864	Topic865	Topic866	Topic867	Topic868	Topic869	Topic870	Topic871	Topic872	Topic873	Topic874	Topic875	Topic876	Topic877	Topic878	Topic879	Topic880	Topic881	Topic882	Topic883	Topic884	Topic885	Topic886	Topic887	Topic888	Topic889	Topic890	Topic891	Topic892	Topic893	Topic894	Topic895	Topic896	Topic897	Topic898	Topic899	Topic900	Topic901	Topic902	Topic903	Topic904	Topic905	Topic906	Topic907	Topic908	Topic909	Topic910	Topic911	Topic912	Topic913	Topic914	Topic915	Topic916	Topic917	Topic918	Topic919	Topic920	Topic921	Topic922	Topic923	Topic924	Topic925	Topic926	Topic927	Topic928	Topic929	Topic930	Topic931	Topic932	Topic933	Topic934	Topic935	Topic936	Topic937	Topic938	Topic939	Topic940	Topic941	Topic942	Topic943	Topic944	Topic945	Topic946	Topic947	Topic948	Topic949	Topic950	Topic951	Topic952	Topic953	Topic954	Topic955	Topic956	Topic957	Topic958	Topic959	Topic960	Topic961	Topic962	Topic963	Topic964	Topic965	Topic966	Topic967	Topic968	Topic969	Topic970	Topic971	Topic972	Topic973	Topic974	Topic975	Topic976	Topic977	Topic978	Topic979	Topic980	Topic981	Topic982	Topic983	Topic984	Topic985	Topic986	Topic987	Topic988	Topic989	Topic990	Topic991	Topic992	Topic993	Topic994	Topic995	Topic996	Topic997	Topic998	Topic999	Topic1000
Smith	0.237	0.346				0.08							0.029																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																																											



**EXCEL: OPEN  
FOR\_EXCEL\_ALL.CSV**



**GEPHI: LAUNCH IT, LOAD THE GDF FILE  
OPEN GEPHITOSIGMAJS\_MIXED.PDF**

# **GETTING YOUR WEBSITE UP**

# Export from Gephi

- Under Export, if you have the plugin installed, you should see an option for SigmaJs. Do that...
- It will create a directory called “network.”
- The files need a little preprocessing...

Run:

```
>python run_network.py network 8000
```

**REVIEW YOUR SITE!**

# How can we improve on the results?

Iterate on number of topics you output. Try 10, instead of 15!

Rerun the GUI with 10 topics, then use command line:

- > cd files

- > python make\_gephi\_file.py topic\_output/output\_csv 10

# How else can we improve?

Pre-process the documents — change what's modeled! Maybe only verbs?

- Use stop words tuned for your data set (don't want proper nouns? or only proper nouns?)
- Read in a document, parse it, save out a new “document” of the POS you want, then use those in the topic modeler

# Cheat sheet: Scripts workflow

1. Optionally: Pre-process text files for part of speech...  
>python preprocess\_files.py [original\_dir] [new\_dir] [part\_of\_speech]
2. Run Mallet on text files.  
Make an output directory. Double click on TopicModelingTool.jar.
3. Create gephi gdf files  
>python make\_gephi\_file.py [topics\_dir\_csv\_path] [optl label]
4. View in Gephi, fix layout....  
Click on Gephi app and open your .gdf output file.
5. Optionally: Output gexf / json for d3  
>python d3\_gexf.py [your output 'for excel' csvfile from 3] [label]
6. From Gephi, export as sigma.js site  
Find under export menu, if you installed that plugin
7. Fix up the sigma config and run server  
>python run\_network.py [network\_dir] [optl port]

Optionally: Preprocess the text before modeling.

## **PRE-PROCESS FOR PARTS OF SPEECH, ETC**



# Our next step... Preprocess the doc text

Use the notebook POS\_Text\_Conversion.ipynb

In this notebook, we'll look at how to handle text: tokenize it, clean it, strip out words/punctuation, find parts of speech...

For faster, command-line use (requires pattern and nltk installed!)

```
>cd files
```

```
>python preprocess_files.py ../data/mixed_chapters ../data/verbs_only VB
```

# Now look at those results...

1. Make a directory for the new topic modeled files under files:

```
> mkdir verb_output
```

2. Rerun the GUI with this as output directory, and the verb files as your input files!

3. Then use command line:

```
> cd files
```

```
> python make_gephi_file.py verb_output/output_csv verbs
```

4. Then find the output  
for\_gephi\_topics\_verbs.gdf & visualize in Gephi.

Optionally...

## **OTHER WAYS TO VIEW TOPICS (AND JSON/D3 EXPORT)**

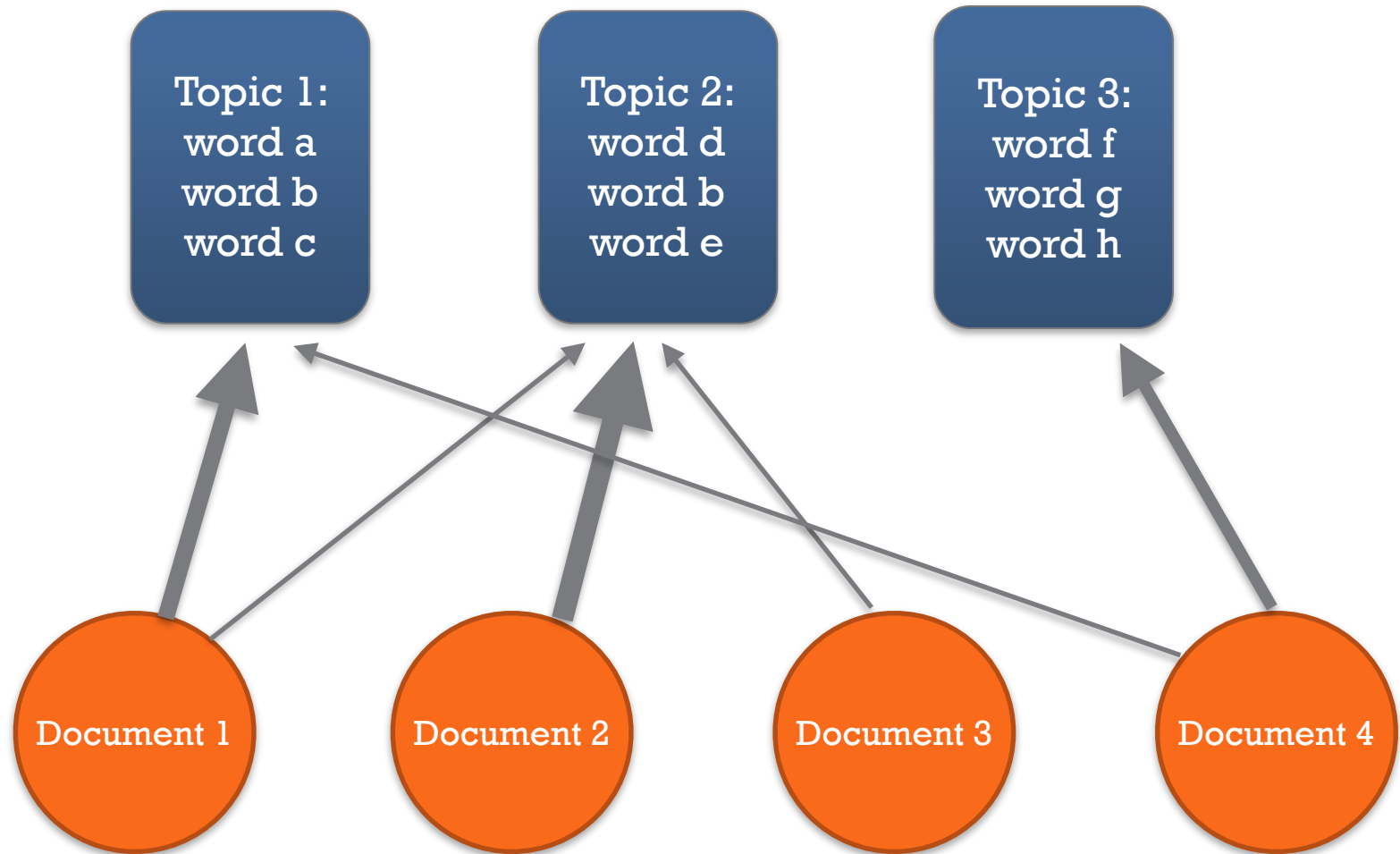
# Cheat sheet: Scripts workflow

1. Optionally: Pre-process text files for part of speech...  
    >python preprocess\_files.py [original\_dir] [new\_dir] [part\_of\_speech]
2. Run Mallet on text files.  
    Make an output directory. Double click on TopicModelingTool.jar.
3. Create gephi gdf files  
    >python make\_gephi\_file.py [topics\_dir\_csv\_path] [optl label]
4. View in Gephi, fix layout....  
    Click on Gephi app and open your .gdf output file.
5. Optionally: Output gexf / json for d3  
    >python d3\_gexf.py [your output 'for excel' csvfile from 3] [label]
6. From Gephi, export as sigma.js site  
    Find under export menu, if you installed that plugin
7. Fix up the sigma config and run server  
    >python run\_network.py [network\_dir] [optl port]

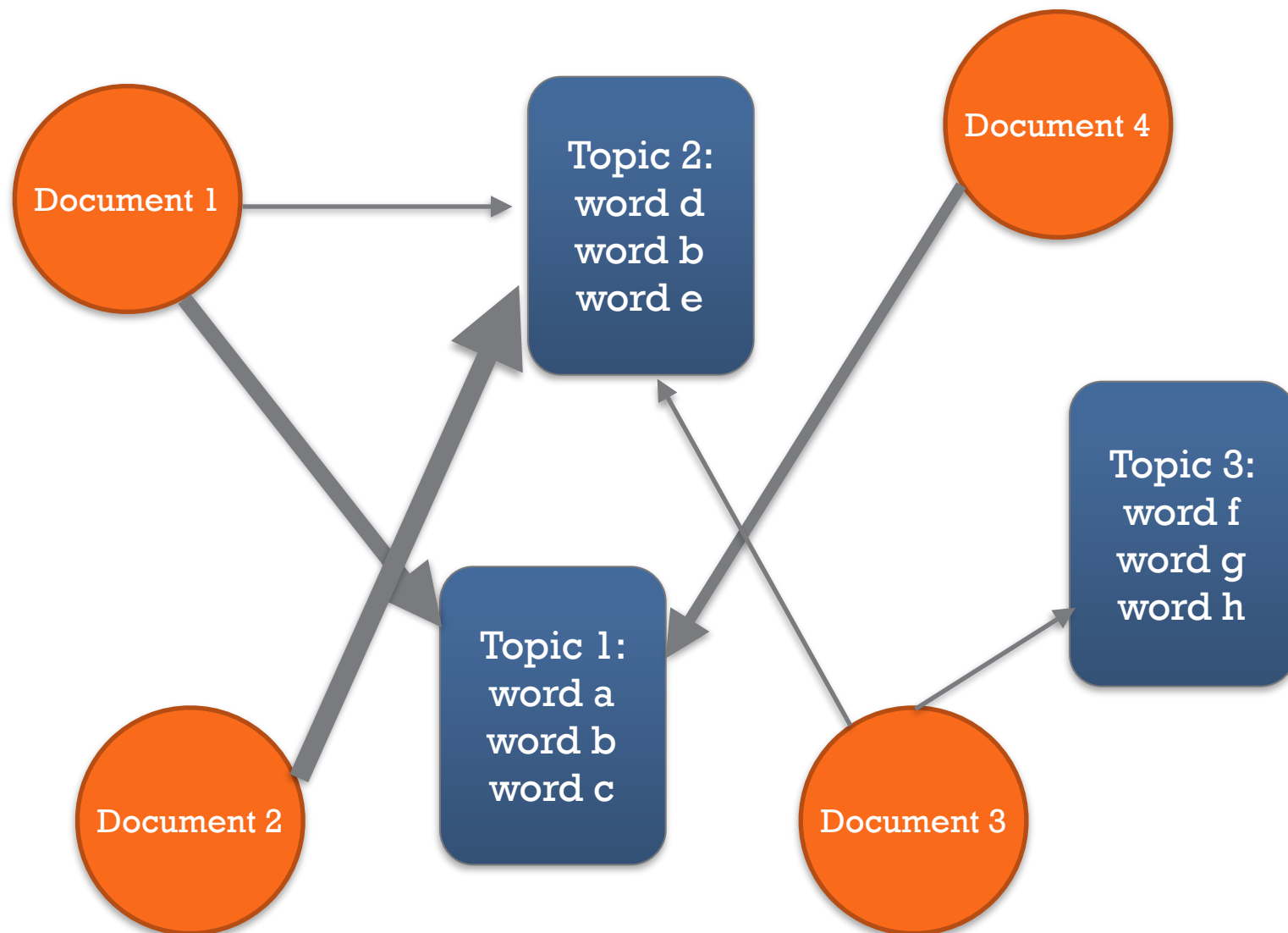
# Going to D3

- Raw nodes-edges json:
  - export json from gephi (using JSON plugin)
  - or post-process and create the json: see next step
  - Example of d3 network use: <http://bl.ocks.org/mbostock/4062045>
- Export gexf and use Elijah Meeks' code to process and display it from gexf format:
  - <http://bl.ocks.org/emeeks/9357371>

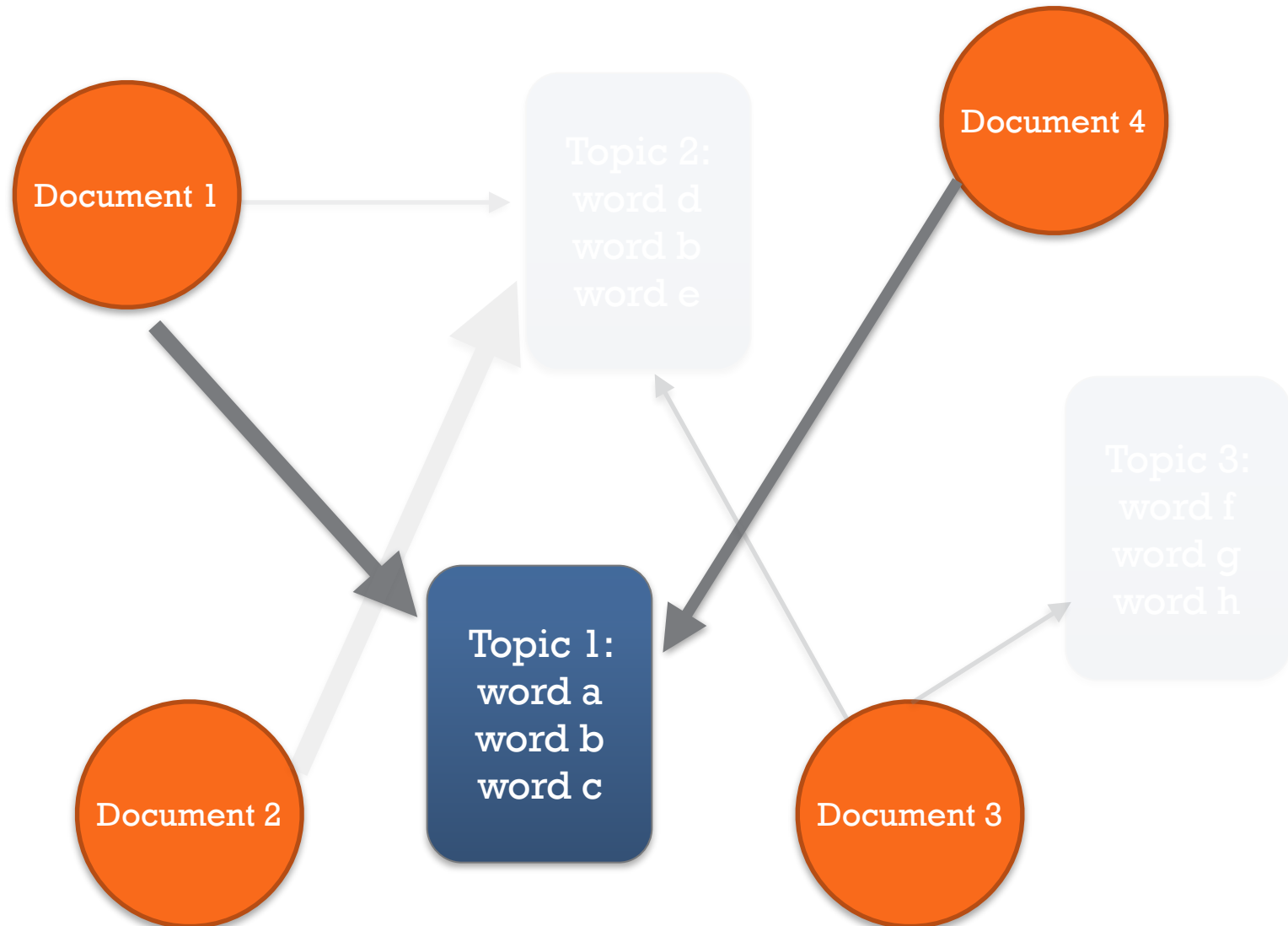
# How we started...



# Another view: More Like Ours



# Another view: Related Document to Document





# But we need to account for link weights

- Doc A — Topic 1, 40% words
- Doc B — Topic 1, 20% words
  
- How do you compute the weight of the relation between the Doc A and Doc B?
  - Options:  $1 / \text{difference}$  (normalized)
  - Average, Median, Count of edges

# Our next step... Advanced: Doc to Doc Network in D3.

We want to combine some of the output we created as JSON files.

Use code in Advanced-D3 and GEXF Network of Docs Only.ipynb or files/d3\_gexf.py.

From the command line - to make a simple file:

```
>cd files
```

```
>python d3_gexf.py for_excel.csv all for_excel_verbs.csv verbs
```

To make a file with 2 sets of edges, to compare:

```
python d3_gexf.py for_excel.csv all for_excel_verbs.csv verbs
```

# Output

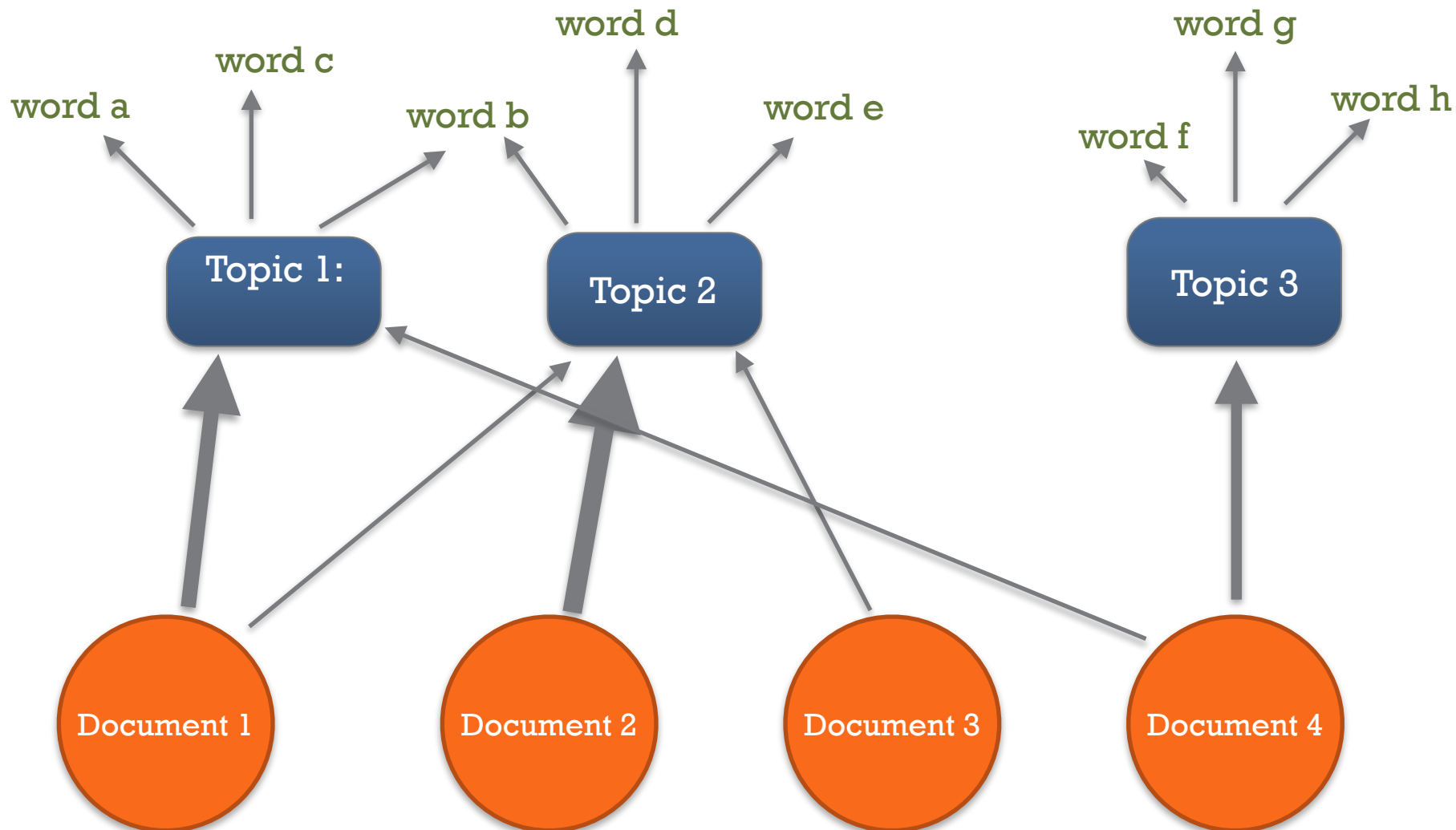
- gexf file — for use in gephi if you want
- json files for use in d3
- Note: combined json, if you input 2 csv files to compare. Don't forget you need to include labels after each csv filename!

```
>cd verb_output
```

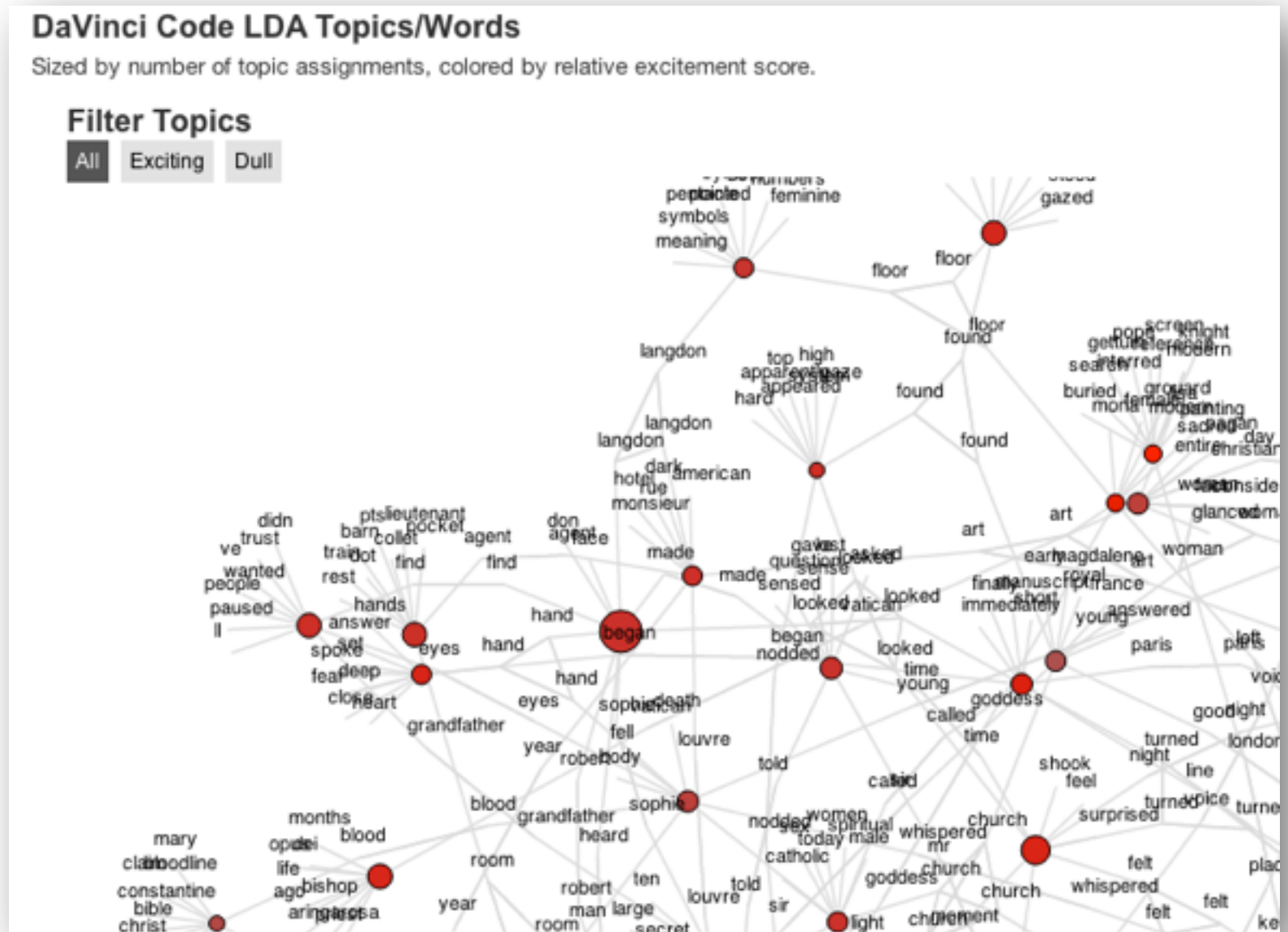
```
>python -m SimpleHTTPServer 8010
```

Load the d3\_network.html file.

# A further level of network you could draw....



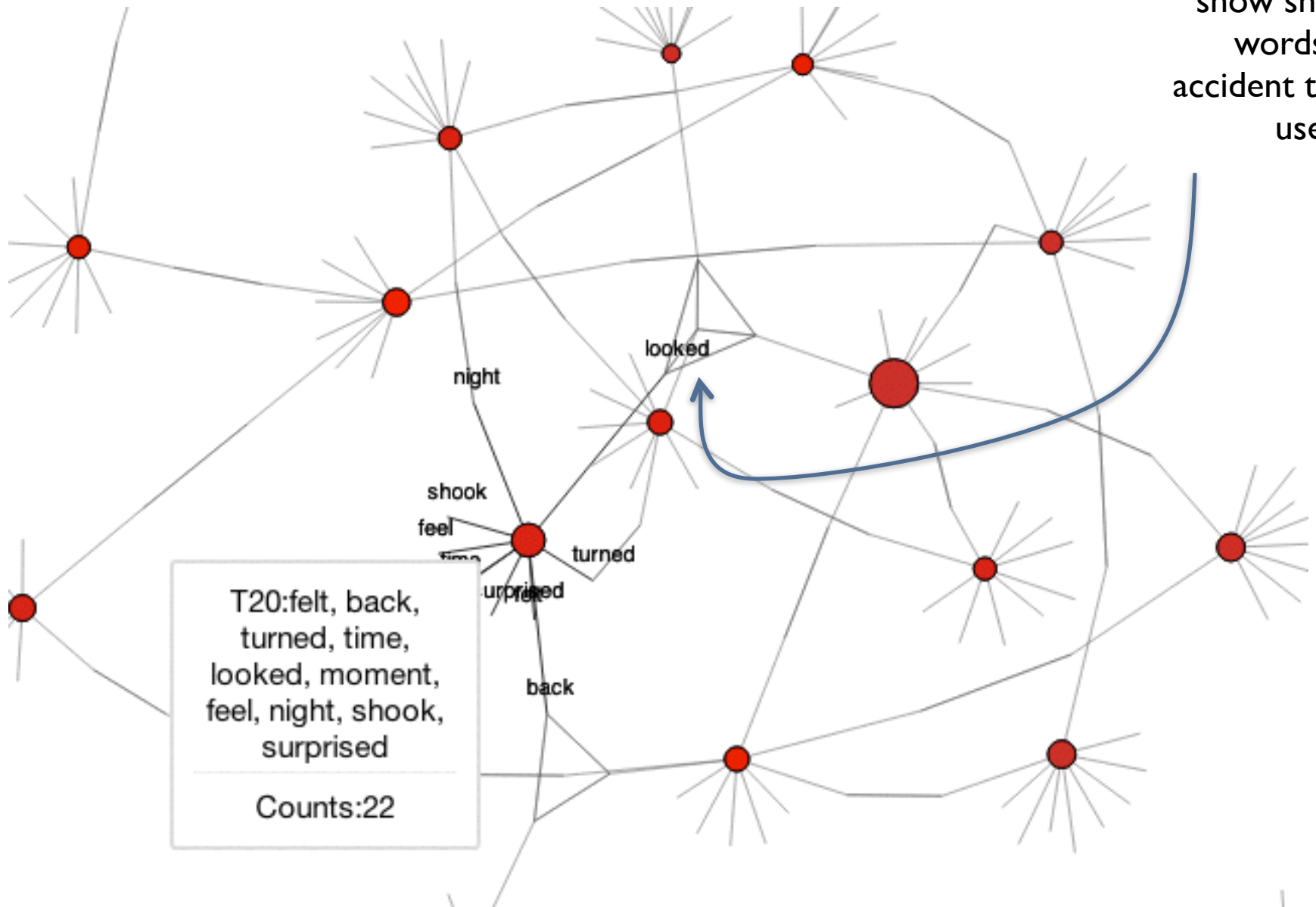
Maybe I need One More Tool. Any word relations of interest?  
Let's try another hairball...



Demo: [http://www.ghostweather.com/essays/talks/openvisconf/topic\\_words\\_network/index.html](http://www.ghostweather.com/essays/talks/openvisconf/topic_words_network/index.html)

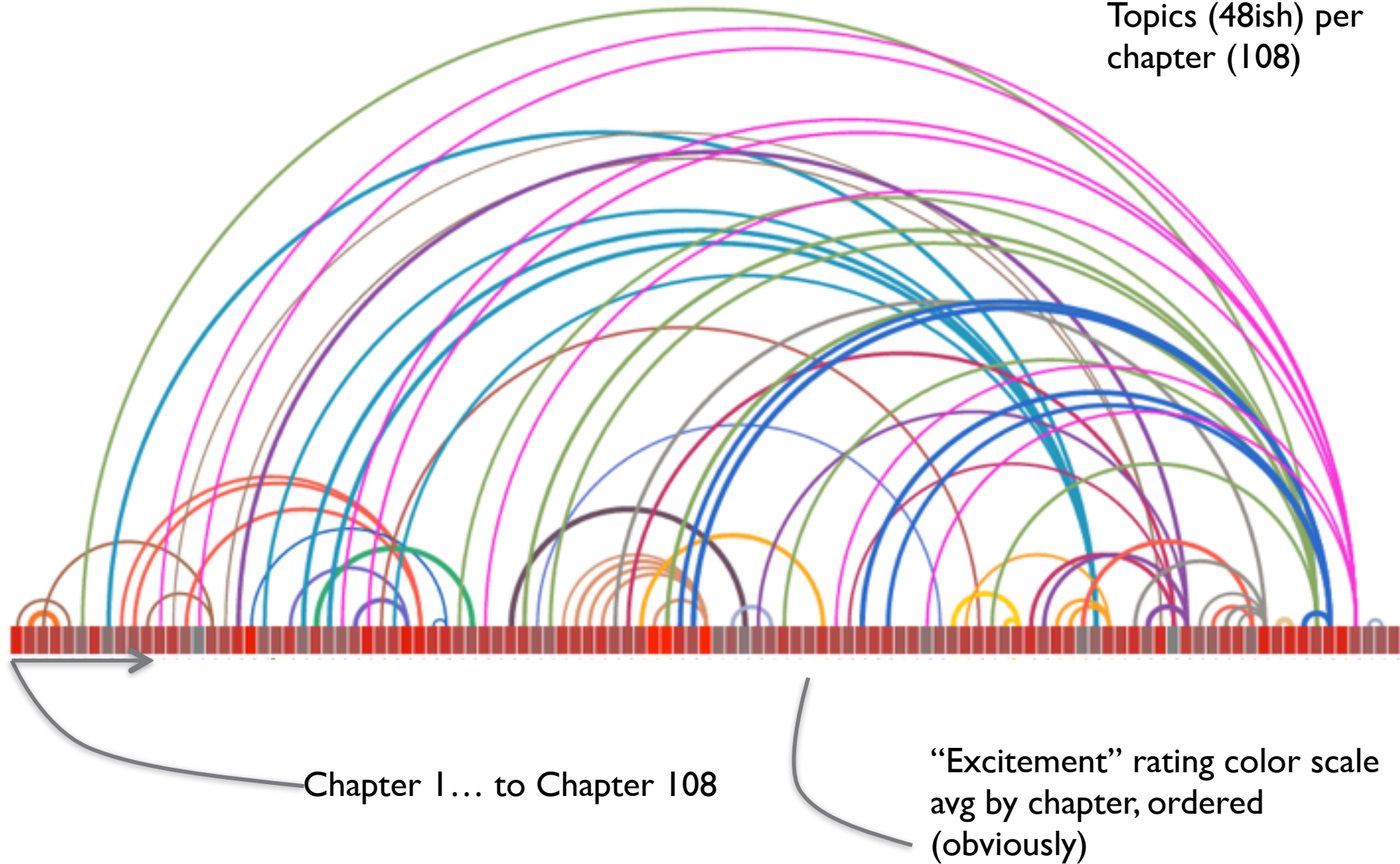
Filtered to only the  
“exciting” nodes...

Small  
“constellations”  
show shared  
words (an  
accident that’s  
useful!)



Another tool: Sequential documents, with topic arcs.  
DaVinci Code topics to chapters mapping

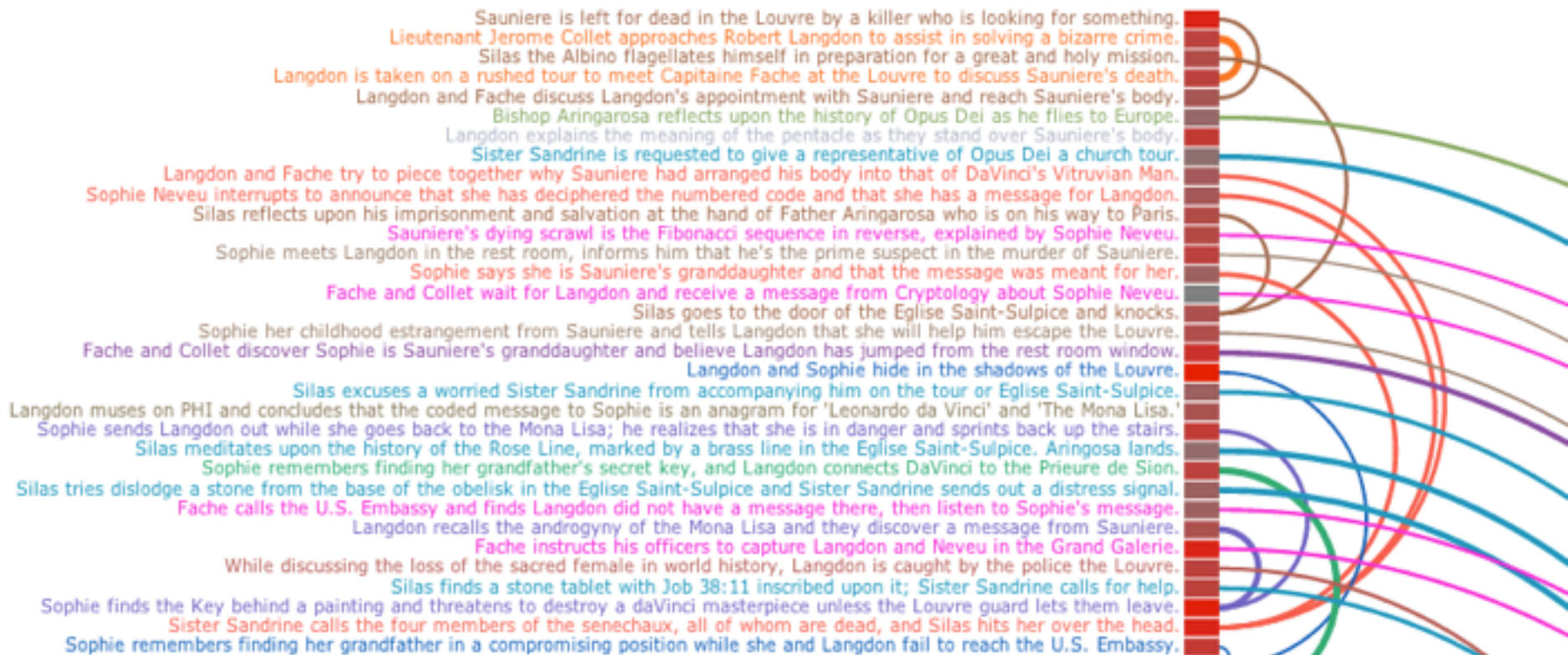
Topics (48ish) per  
chapter (108)



Chapter 1... to Chapter 108

“Excitement” rating color scale  
avg by chapter, ordered  
(obviously)





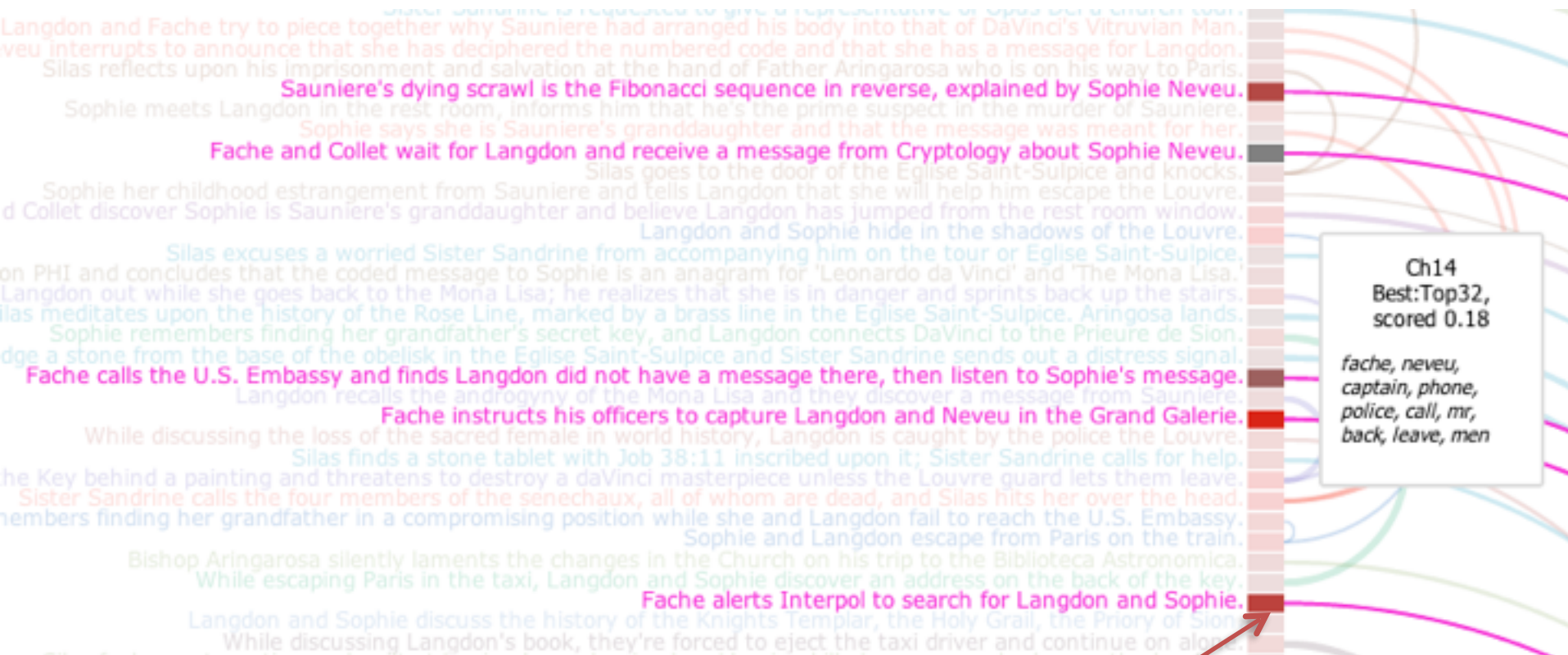
Add some topic-tooltips  
and fade-outs....





# But what did this show?

Some topics are just neither exciting nor dull – topic clustering (as I did it) had little to do with action scenes. It's slightly helpful for topics, though ☺



These nodes are shaded from  
gray (dull) to red (exciting)

**WRAP UP!**

# Other Ways to Improve the Results Display

- Visualize differently or more (chords, matrix...)
- Look for the topic words “in context” - find sentences with them and use those as part of your topic description
- Construct phrases from your topic words to make them “better” for descriptors
- Use only the interesting output words for a topic
- Don’t use the result immediately — use as input to other methods (it’s a data reduction technique like principal components analysis)

# A Few More References

- Matthew Jockers' post: The LDA Buffet is Now Open: <http://www.matthewjockers.net/2011/09/29/the-lda-buffet-is-now-open-or-latent-dirichlet-allocation-for-english-majors/>
- Scott Weingart's nice overview of LDA Topic Modeling in Digital Humanities: <http://www.scottbot.net/HIAL/?p=221>
- Elijah Meeks' lovely set of articles on LDA & Digital Humanities vis: <https://dhs.stanford.edu/comprehending-the-digital-humanities/>
- Topic Modeling Made Just Simple Enough - Ted Underwood post: <http://tedunderwood.com/2012/04/07/topic-modeling-made-just-simple-enough/>
- Some pure python (and C) implementations (toy code, primarily) are listed on Blei's website: <http://www.cs.princeton.edu/~blei/topicmodeling.html>