2017

The Document Similarity Network: A Novel Technique for Visualizing Relationships in Text Corpora

Dylan Baker
Harvey Mudd College

The Document Similarity Network: A Novel Technique for Visualizing Relationships in Text Corpora

Dylan Baker

Talithia Williams, Advisor

Tanja Srebotnjak, Reader
Blake Hunter, Reader

HARVEY MUDD COLLEGE

Department of Mathematics

May, 2017

Abstract

With the abundance of written information available online, it is useful to be able to automatically synthesize and extract meaningful information from text corpora. We present a unique method for visualizing relationships between documents in a text corpus. By using Latent Dirichlet Allocation to extract topics from the corpus, we create a graph whose nodes represent individual documents and whose edge weights indicate the distance between topic distributions in documents. These edge lengths are then scaled using multidimensional scaling techniques, such that more similar documents are clustered together. Applying this method to several datasets, we demonstrate that these graphs are useful in visually representing high-dimensional document clustering in topic-space.

Contents

List of Figures

List of Tables

Acknowledgments

In addition to my incredible advisor, Prof. Talithia Williams, I would also like to thank Prof. Weiqing Gu, and my readers Prof. Tanja Srebotnjak and Prof. Blake Hunter, for their guidance and support.

Chapter 1

Introduction

Finding similarities between written documents in large bodies of text is useful; one might want to, for instance, automatically distinguish spam emails from legitimate ones, or group news articles by their subject. Clustering, a machine learning technique used to find groupings within datasets based on shared characteristics, is an effective way of accomplishing this goal. However, applying clustering on inherently noisy data, such as text data, often has drawbacks; because it is nearly always done in high-dimensional space, it is incredibly difficult to see if one's clustering algorithm is appropriate or if one's clusters make sense. Thus, visualizing high-dimensional data in fewer dimensions is a helpful first step when performing cluster analysis on data.

We look specifically at visualizing relationships between document topics in text corpora. In order to visualize these connections, we will construct graphs, with nodes representing documents and edges representing similarities between these documents: we call these Document Similarity Networks (DSNs). Because this project combines several distinct processes, it is important to first give a brief overview of the context surrounding them and the work currently being done in the field. Ultimately, constructing a DSN hinges on two main processes: relating documents using their shared topics, and creating the graph itself. In this thesis we demonstrate a method to construct a DSN and test it on several text data sets.

1.1 Topic Extraction

There are two overarching approaches to extracting topics from text corpora: applying basic dimension reduction to a matrix which encodes information

$$\begin{array}{ccccc} Term \times Document & & Term \times Topic & & Topic \times Document \\ V & & W & & H \\ \begin{bmatrix} \cdot&\cdot&\cdot&\cdot&\cdot\\ \cdot&\cdot&\cdot&\cdot&\cdot\\ \cdot&\cdot&\cdot&\cdot&\cdot\\ \cdot&\cdot&\cdot&\cdot&\cdot\\ \cdot&\cdot&\cdot&\cdot&\cdot \end{bmatrix} & \approx & \begin{bmatrix} \cdot&\cdot\\ \cdot&\cdot\\ \cdot&\cdot\\ \cdot&\cdot\\ \cdot&\cdot \end{bmatrix} & \times & \begin{bmatrix} \cdot&\cdot&\cdot&\cdot&\cdot\\ \cdot&\cdot&\cdot&\cdot&\cdot \end{bmatrix} \end{array}$$

Figure 1.1 Illustration of NMF as it applies to topic extraction; a *Term* × *Document* matrix is approximated as the product of a *Term* × *Topic* matrix and a *Topic* × *Document* matrix

about the documents, and applying more sophisticated statistical methods to analyzing the corpus that make more specific assumptions about the data.

1.1.1 Nonnegative Matrix Factorization

Nonnegative Matrix Factorization (NMF) approximates a sparse matrix as the product of two smaller matrices, and can be used to extract topics from a corpus encoded in a sparse matrix referred to as a *Term* × *Document* matrix. Each column in a *Term* × *Document* represents a document, and each row represents a word in the corpus' vocabulary. Each entry counts the number of times a specific word occurs in a specific document. This document can be approximated as the product of two non-negative lower-dimensional matrices. These lower-dimensional matrices can be thought of as capturing the underlying "topics" in the document (Lee and Seung, 2001), where the number of topics K is fixed. More specifically, performing NMF on an $N \times M$ *Term* × *Document* matrix V yields the two matrices: a *Word* × *Topic* matrix W multiplied by a *Topic* × *Document* matrix H. Further, W is for which H encodes an approximation of the *Term* × *Document* matrix— V approximates the underlying K topics, and W approximates how those topics are distributed among the documents. A visualization of this approximation is shown in Figure 1.1.

There are a variety of algorithms used to compute these matrices, all of which initialize W and H to arbitrary matrices of dimension $N \times K$ and $K \times M$, then perform updates to W and H such that the difference between their product and V, the initial *Term* × *Document* matrix, is minimized. The most popular method of approximation is Lee and Seung's multiplicative update method (Lee and Seung, 2001). This method minimizes $||V - WH||$ with the constraint that $W, H > 0$ by alternating multiplying W and H by factors involving V, W and H. Gradient descent (Lin, 2007) and alternating least squares (Paatero and Tapper, 1994) methods are also commonly used (Ho, 2008), which take different approaches to minimizing similar objective functions.

Much work around NMF involves modifying these update steps in the multiplicative update method to promote various qualities in W and H. For example, work in signal processing has brought about semi-supervised NMF that incorporates a regularizing term to the objective function that penalizes deviation from any known data (Lee et al., 2010). This has been shown to improve clustering performance in the context of machine learning. The major drawback of NMF when applied to topic extraction, however, is that it assumes that topics are fixed across texts, which is often an unrealistic expectation of a corpus and thus leads to less human-comprehensible topics (Stevens et al., 2012).

In the following sections, we detail another *Term* × *Document* matrix dimension reduction technique used in topic modeling, then continue on describe the work leading up to the development of the method we chose to use, Latent Dirichlet Allocation.

1.1.2 Probabilistic Latent Semantic Indexing

Probabilistic Latent Semantic Indexing (Hofmann, 1999) is a probabilistic method of topic extraction, one which motivated the development of Latent Dirichlet Allocation by Blei et al. (2003). This method builds on Latent Semantic Indexing (LSI) (Deerwester et al., 1990), which itself relies on the *term frequency-inverse document frequency* (tf-idf) statistic, a common weighting scheme in the area of recommender systems (Beel et al., 2016). Tf-idf, in this context, reflects the relative importance of each word to each document in a given corpus. Within a document, the tf-idf value for each word is proportional to the number of times it appears in that document; all values are then normalized to account for the number of times each word appears in the corpus altogether. One can imagine a *Term* × *Document* matrix, then, with tf-idf values for each word, instead of word frequency. Latent Semantic Indexing applies a matrix dimension-reduction technique similar to NMF, called Singular Value Decomposition (SVD), which finds a linear subspace of the tf-idf *Term* × *Document* matrix that reveals the terms that account for the most variation in the document word distributions. This process is used for very rudimentary synonym detection.

Probabilistic Latent Semantic Indexing (pLSI) is a probabilistic model that builds on LSI. It was initially developed for use in parsing natural language queries to search engines, addressing the issue of the ambiguity and variability in search queries written by internet users. It is a generative model that, in addition to detecting synonyms, is able to detect rudimentary topics. In pLSI, each document is modeled as a mixture model across topics, which are modeled as multinomial random variables. Each word is in each document is thought of as coming from a single topic.

In developing Latent Dirichlet Allocation, Blei et al. (2003) critique pLSI for its large number of parameters (which scales linearly with the size of the corpus and can lead to overfitting), for its assumption that each word belongs to a single topic, and for its inability to assign probabilities to new documents once its parameters have been tuned on a single corpus.

1.1.3 Latent Dirichlet Allocation

Latent Dirichlet Allocation (LDA) is a generative statistical model that introduces slightly more complexity to accomplish the goal of topic extraction. LDA assumes that documents are comprised of distributions of latent topics, which themselves are a mixture over underlying topic probabilities distributions(Blei et al., 2003). Like NMF, LDA assumes a fixed and known underlying number of topics. LDA also rests on the assumption that the order of documents, and the order of words within documents, are both irrelevant– LDA treats each document as a distinct "bag of words". While models have been developed building on LDA that take into account word and document order, a bag-of-words approach has yielded suitable topics for this and similar applications (such as NMF). Further, standard LDA is widely-used, and fast, open-source implementations make it easy to use in this context.

Overall, LDA is useful in that it tends to produce significantly more coherent topics and tends to be more well-suited to analyzing text corpora than basic dimension-reduction techniques. Using LDA to extract topics for document clustering has been used in the past. For example, in their application *iVisClustering*, an "interactive visual analytics system for document clustering", Lee et al. (2012) chose to use LDA as its baseline topic-extraction model. Similarly, LDA has been used for grouping text by sentiment in opinion mining (Moghaddam and Ester, 2011), as the base upon which a generative clustering algorithm that takes into account both text and tag data on social bookmarking websites (Ramage et al., 2009), and to improve the performance of other clustering algorithms (Chen et al., 2013).

Overall, LDA consistently produces more cohesive topics than the methods described previously, and, as we have detailed, is popular in related applications. Because topic extraction is central to the effective construction of DSNs, and LDA is arguably the most effective way to extract topic distributions from documents, we ultimately chose to use the smoothed LDA model for topic extraction. We will describe the more formal definition of LDA in Chapter 2.

Next, we will describe how we can use this topic extraction to measure the similarities between documents and subsequently use these similarities to construct our Document Similarity Networks.

1.2 Graph Construction

Once topics are extracted from each document, we construct a graph whose edges are scaled to represent the similarity between documents. To do this, we need to first establish a definition for document similarity and a method for scaling edge lengths. Document similarity is here defined using the distance between documents topic distributions, and edge lengths are determined using multidimensional scaling.

1.2.1 Distance Metric

Because we're using LDA to extract distributions of topics from individual documents, we need a means of determining the distance between two topic distributions. There are a variety of metrics used to calculate the similarity between two distributions; the most commonly-used metrics in this context are types of $f-$divergence, such as Kullback-Leibler divergence and Hellinger Distance. Many distribution distance metrics would be appropriate to use in this case; we ultimately chose to use Hellinger distance because it has been used to measure the distance between topic distributions in comparable work (Blei and Lafferty, 2007).

1.2.2 Edge Length Scaling

Edge lengths were scaled using metric multidimensional scaling. Because we first calculate the Hellinger distances between each pair of documents, we can treat each document as a point in 2-dimensional space and assign each point a location such that points that are "far away" from each other (that is, they correspond to documents with larger Hellinger distances between them) are placed further apart in two-dimensional space. This is done by minimizing an objective function referred to as "stress", which we will describe in more detail in Chapter 2. Multidimensional scaling refers to the calculation of these points through this loss function minimization. Consequently, when edges are drawn between these points, they are scaled to represent the distances between document topic distributions.

1.3 Summary

We have now detailed our approach to assist document cluster analysis by presenting a two-dimensional visualization of document similarities called a Document Similarity Network (DSN). To construct each DSN, we start by using Latent Dirichlet Allocation to extract topic distributions from documents in a text corpus. We then compute the Hellinger distances between these topic distributions. From there, we construct a graph with nodes representing each document. We draw edges between these nodes, whose lengths are scaled using multidimensional scaling such that documents that are closer in topic-space are drawn closer together. In Chapter 2, we will describe each of these steps in more technical detail. We will then describe how we implemented our approach on two very different datasets in Chapter 3. We show and discuss our results in Chapter 4, and summarize our findings in Chapter 5.

Chapter 2

Background

We will now explore in more detail the process of constructing a DSN. First, we outline a definition of variables and define the probability of the Latent Dirichlet Allocation model. We then give the formal definition of Hellinger distance, followed by a brief explanation of multidimensional scaling as it is applied here. Finally, we show an example of edges in a small DSN to illustrate how these methods are combined.

2.1 Extracting Topics

To extract topics from the corpus, we use smoothed Latent Dirichlet Allocation (LDA) (Blei et al., 2003). LDA is a generative statistical model that models corpora as follows: documents in a corpus are comprised collections of words that have been sampled from a Dirichlet distribution of topics specific to that document, where each of these topics is a Dirichlet distribution of words over the vocabulary of the corpus. Specifically:

- A vocabulary V is a set containing every unique word w_i contained in the corpus.
- A corpus is comprised of M documents, each with N_m words. These words are not ordered.
- We assume there are K underlying topics. Each topic φ is a Dirichlet distribution (with parameter β, a vector of positive real values) of words from the vocabulary V. These topics are denoted $\varphi_{1:K} \sim \mathrm{Dir}(\beta)$.

- We assume there are underlying Dirichlet distributions of topics for each document (with parameter α, a vector of positive real values). These are denoted $\theta_{1:M} \sim \mathrm{Dir}(\alpha)$.
- In performing LDA, for each word in each document, we estimate which topic it was sampled from; these estimated topics are denoted $z_{1:M,1:N}$

The overall probability of the LDA model can be expressed:

$$P(\boldsymbol{W}, \boldsymbol{Z}, \boldsymbol{\theta}, \boldsymbol{\varphi}; \alpha, \beta) = \prod_{i=1}^{K} P(\varphi_i; \beta) \prod_{j=1}^{M} P(\theta_j; \alpha) \prod_{t=1}^{N} P(Z_{j,t}|\theta_j) P(W_{j,t}|\varphi_{Z_{j,t}})$$

An intuitive visual depiction of this model is given in Figure 2.1, which depicts the smoothed LDA model in plate notation. Plate notation is a graphical way to draw out hierarchical Bayesian models with repeated variables. Each "plate" (or rectangle) represents a repeated variable, and each arrow represents a dependency under which the repeated variables are assumed to be independent and identically distributed (iid)— to take an example from Figure 2.1, each of the K topics $\varphi_{1:K}$ are assumed to be iid, given the underlying Dirichlet parameter β.

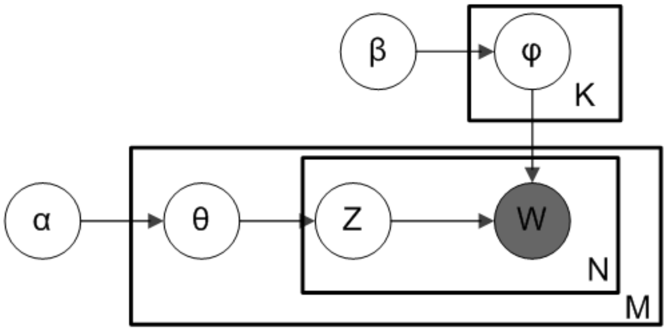

Figure 2.1 Plate notation for smoothed LDA (Slxu, 2009)

2.2 Measuring Document Similarity

We used Hellinger distance, a form of f−divergence, to measure the similarity between two document topic distributions. For two discrete distributions $P = (p_1, \ldots, p_k)$ and $Q = (q_1, \ldots, q_k)$, the Hellinger distance H is defined as:

$$H(P,Q) = \frac{1}{\sqrt{2}}\sqrt{\sum_{i=1}^{k}(\sqrt{p_i} - \sqrt{q_i})^2}$$

We defined the similarity between two documents P, Q as simply $1 - H(P,Q)$.

2.3 Multidimensional Scaling

The nodes in each DSN are placed using multidimensional scaling (MDS) (Wasserman and Faust, 2009), a form of non-linear dimensionality reduction that assigns each node in our graph a location in two-dimensional space such that more similar nodes are placed closer together. The implementation of MDS used here minimizes the loss function of "stress", a residual sum of squares using the Euclidean distance between points.

2.4 Example

In Chapter 3 we'll examine in more detail the process of constructing a graph; however, in Figure 2.2 we show a broader overview of how a DSN is constructed. Similarities between the topic distributions of documents are calculated (recall that similarity here is defined simply as (1− Hellinger distance) and are used to scale the distances between nodes in a graph, each of which represent a document in a corpus. While edges that represent significantly different documents are not drawn, they are still used to scale the distances between nodes and are simply removed to make the DSN more readable.

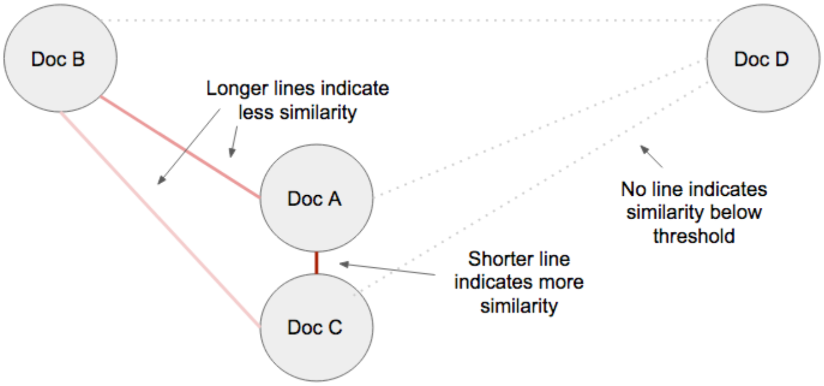
Doc B
Doc D
Longer lines indicate less similarity
Doc A
No line indicates similarity below threshold
Shorter line indicates more similarity
Doc C

Figure 2.2 Sample DSN. The length of each edge is proportional to the Hellinger distance between the topic distributions of each document. A similarity threshold is set below which no edge will be drawn; this is simply to improve the clarity of DSNs containing many documents.

Chapter 3

Methods

3.1 Datasets

We've initially looked at two datasets. First, we used a dataset consisting of 2,225 BBC News articles published from 2004-2005 corresponding to five topical areas of news: business, entertainment, politics, sport, and tech (Greene and Cunningham, 2006). Because this dataset has labels, we're able to see the efficacy of our model on showing clustering with already-categorized documents. Second, we used a subset of roughly 1 million comments posted in May of 2015 on the media-aggregation and content-sharing website Reddit (Reddit, 2015) across 60 "subreddits", which are sub-forums centered around different shared interests.

3.2 Data Pre-processing and Network Construction

All text data was preprocessed before running LDA: punctuation was removed, words were converted to lowercase and stemmed, and stop words were removed. For the Reddit data, all comments from each subreddit were concatenated into a single string and used to construct a *Term* × *Document* matrix across subreddits. For the BBC data, all documents across the 5 topics were processed together. For each corpus, a vocabulary was constructed mapping each unique word across texts to an index. Each document was then represented as a list of tuples listing the index of each word in the document and the number of times that word occurred. An LDA model of the data was constructed using this vocabulary, the collection of documents represented as lists of tuples, and our fixed number of topics K as parameters;

this was done using the `gensim` python library.

Next, each document's topic distribution was converted to a vector with K entries, each entry representing the proportions of each topic present in each document. The similarity between every pair of documents was calculated, using the Hellinger distance between each of these vectors, and stored in an $M \times M$ matrix.

Lastly, a weighted graph was constructed using the `networkx` library, each node corresponding to a document, with edges between every pair of nodes whose weight was equal to the similarity between those two documents. This graph was saved in GEXF format and imported into the graph visualization platform Gephi. From there, the distances between nodes were calculated using Metric MDS, down-weighting large distances (lower similarities) and up-weighting smaller ones (greater similarities). Edges were also colored based on their weight, with redder edges corresponding to more similar documents. In some cases, nodes were colored based on their source.

3.3 Software Implementation

All code was written in python using Jupyter Notebook (Pérez and Granger, 2007). Words were made lower-case and stemmed using the `nltk` library (Bird et al., 2009), and stop words were removed using the `stop_words` library. The `gensim` library was used for conducting LDA, computing Hellinger distances between each document, and extracting the most prominent topic from each document (Řehůřek and Sojka, 2010). The `networkx` library, which is backed by `matplotlib` (Hunter, 2007), was used for constructing the graph data (Hagberg et al., 2008). The visualization platform Gephi was used for creating the graph visualizations (Bastian et al., 2009); the multidimensional scaling was calculated using the the MDS Statistics and MDS Layout Gephi plugins (Group, 2009). Our code has been posted publicly on GitHub (Baker, 2017).

Chapter 4

Results and Discussion

Next, we show several applications of DSNs. We constructed DSNs with 5, 10, and 50 topics with the BBC News data, and DSNs with 10 and 30 topics with the Reddit comment data. We have found that, for suitably few topics, DSNs are useful in visualizing clustering in topic-space. We now present these results alongside our observations and analysis.

A two-dimensional representation of high-dimensional space is less likely to show high-dimensional clustering due to the amount of information lost in dimension reduction. We see this illustrated when moving from 5 to 50 topics on the BBC news articles in Figures 4.1 and 4.3, respectively. While the 50 topics extracted (given in Tables 4.3 and 4.4) are nearly all useful, the resulting DSN would be difficult to interpret without labels.

When we consider Figure 4.3 with labels, we see that each of the 5 groups are nearly entirely grouped together in space, which indicates that there is likely some high-dimensional clustering. However, without knowing these labels beforehand, it would be nearly impossible to see this clustering by eye (which is the intended purpose of constructing a DSN). In contrast, with 5 topics, while it is unlikely that a user would be able to reconstruct the original five groups of documents without the labels, it is clear that there *is* clustering. Thus, we have found the greatest success on relatively few ($\leq$10) topics.

Consistent with our observation that DSNs are less effective with a larger number of topics, we've found that as the number of topics increases, the overall stress of the DSN increases. However, in coloring the edges based on their similarity, we're able to see where information has been lost in representing the $K-$dimensional data in two dimensions. Where there are longer red edges, we see that two similar documents were unable to be

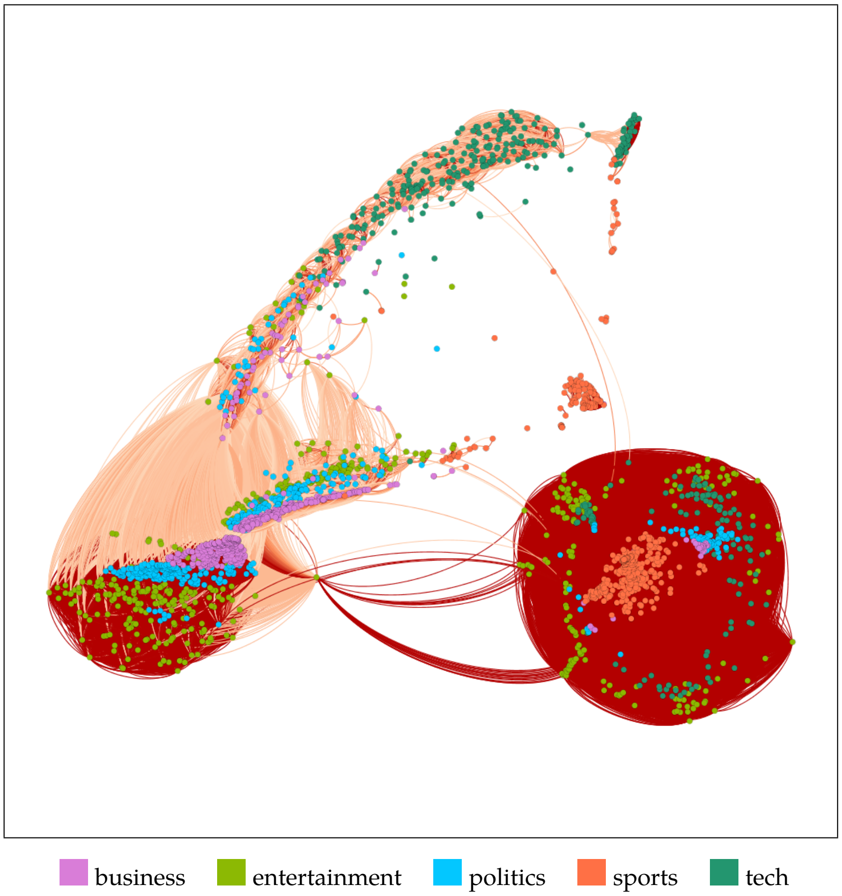

Figure 4.1 2225 BBC news articles with $K = 5$ topics. Darker red lines indicate higher similarity. Scaling stress = 0.20. Edges representing similarity < 0.90 not drawn. Topics, and their approximate corresponding categories, are shown in Table 4.1

.

drawn close together, and thus their similarity was not able to be effectively shown in the DSN. This gives us immediate sense of *where* stress is occurring in the DSN and where clustering is being shown more (or less) effectively. Therefore, we can rely both on our stress value and the edge coloring to indicate whether our DSN is likely to be useful.

We have also found that DSNs are best suited to relatively small corpora. With fewer edges to scale, there is less overall stress on the network, and thus

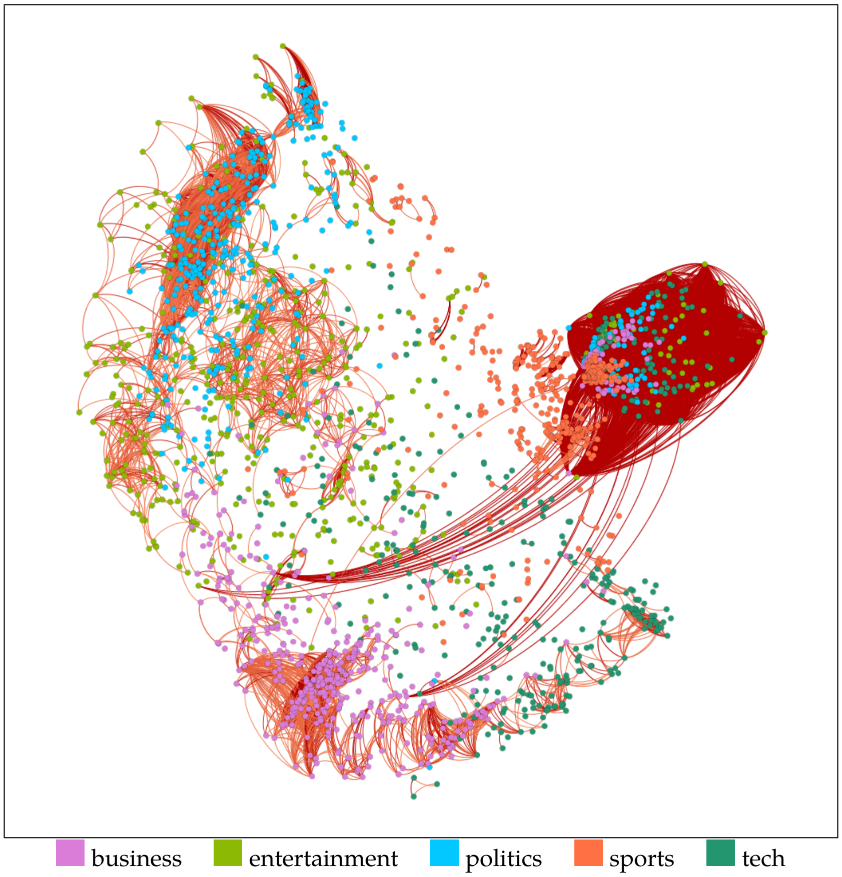

Figure 4.2 2225 BBC news articles with $K = 10$ topics. Darker red lines indicate higher similarity. Scaling stress = 0.230. Edges representing similarity < 0.90 not drawn. Topics, and their approximate corresponding categories, are shown in Table 4.2

.

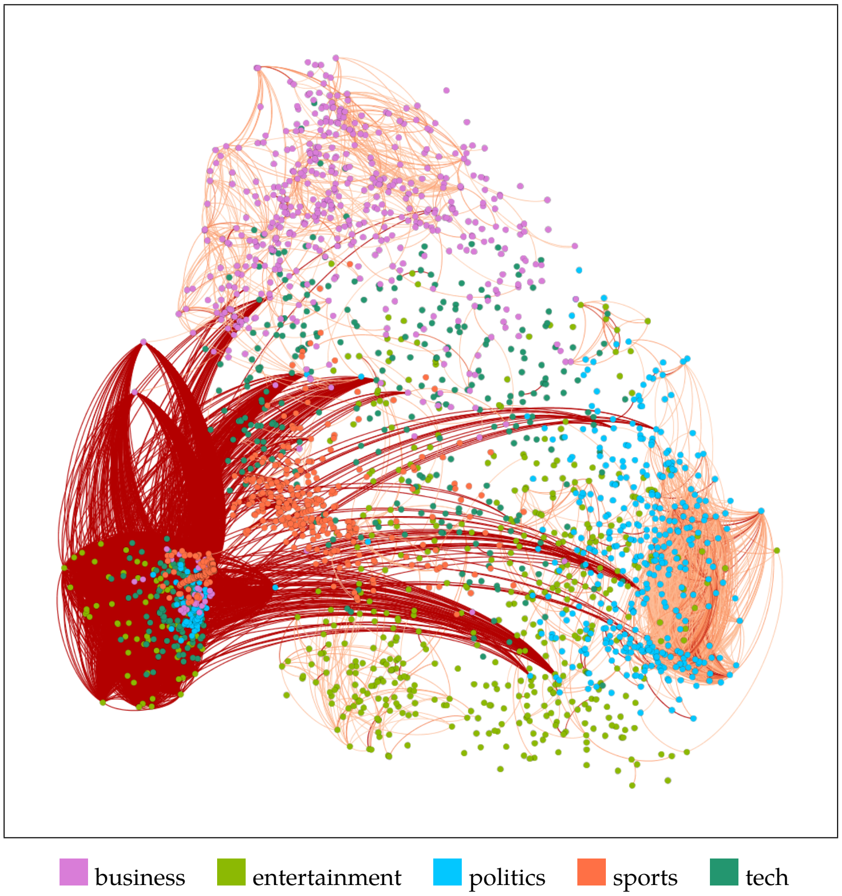

Figure 4.3 2225 BBC news articles with $K = 50$ topics. Darker red lines indicate higher similarity. Scaling stress = 0.27. Edges representing similarity < 0.70 not drawn. Topics listed in Tables 4.3 and 4.4

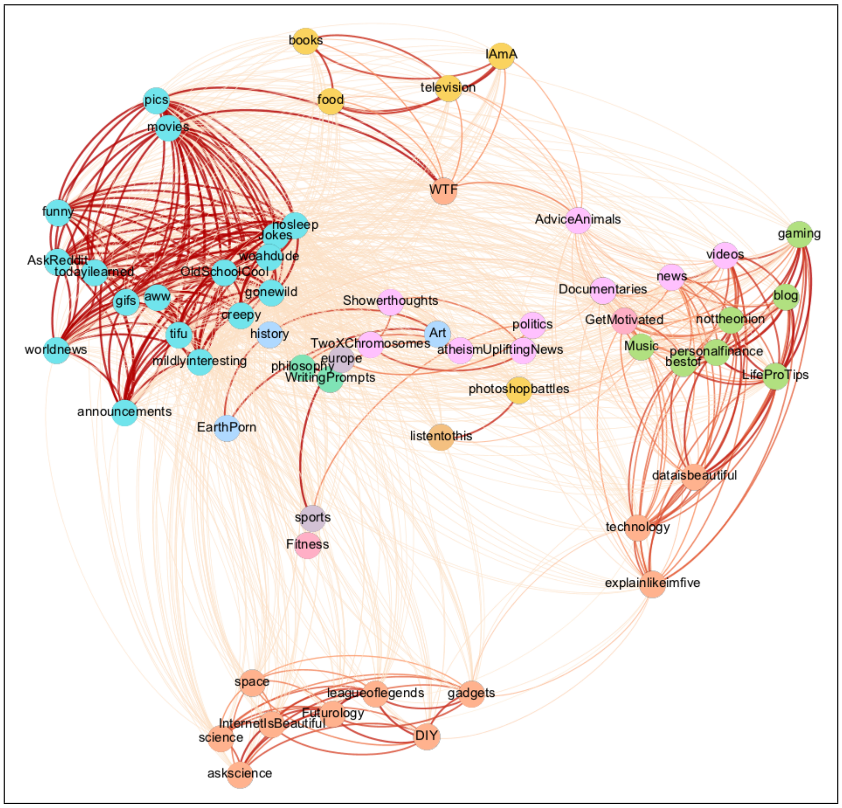

Figure 4.4 60 subreddits, each represented with 1,000 comments, with $K = 10$ topics. Nodes colored by most prominent topic. Darker red lines indicate higher similarity. Scaling stress = 0.179. Edges representing similarity < 0.25 not drawn. Topics shown in Table 4.5.

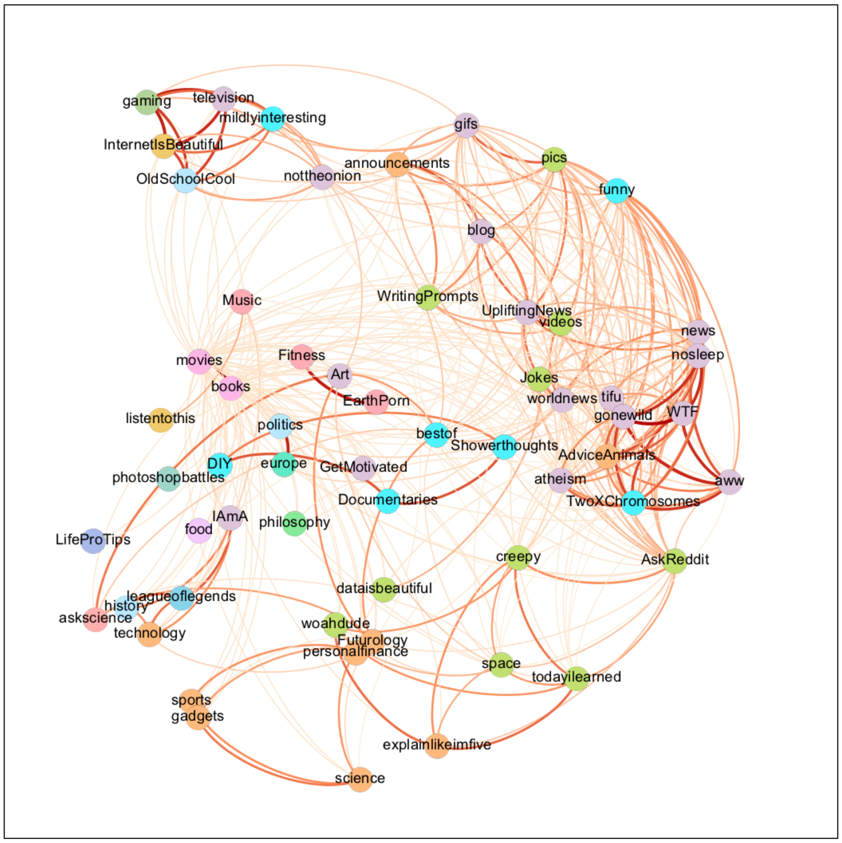

Figure 4.5 60 subreddits, each represented with 1,000 comments, with K = 30 topics. Nodes colored by most prominent topic. Darker red lines indicate higher similarity. Scaling stress = 0.179. Edges representing similarity < 0.25 not drawn. Topics shown in Table 4.6.

Table 4.1 List of 5 topics from BBC News data

business/tech	mobil, technolog, phone, servic, digit, music, tv, new, devic, video, player, market, broadband, mr, million, uk, show, offer, consum, comput
entertainment	film, music, dvd, best, award, us, star, pp, new, first, includ, sale, show, movi, soni, technolog, industri, releas, top, ray
politics	mr, us, govern, new, last, two, first, world, minist, elect, told, parti, labour, plan, win, countri, play, nation, compani, week
sports/online gaming	game, play, world, gamer, spam, titl, mail, consol, new, onlin, domain, hour, player, xbox, life, hd, mr, real, mani, highc
tech	softwar, user, system, firm, net, comput, file, secur, site, search, compani, program, network, mr, microsoft, attack, inform, viru, servic, mani

Table 4.2 List of 10 topics from BBC News data

search, googl, web, yahoo, user, file, site, engin, inform, ask, system, help, desktop, mr, blog, jeev, technolog, new, websit, version	mobil, phone, servic, network, attack, data, technolog, handset, camera, firm, system, compani, user, net, comput, site, oper, custom, number, call
game, player, technolog, new, devic, digit, market, soni, pc, high, music, show, gadget, tv, video, dvd, mr, consol, comput, develop	site, us, robot, mr, softwar, imag, websit, map, research, group, game, action, pirat, read, piraci, pictur, bittorr, news, file, copi
film, best, award, first, star, includ, top, new, name, show, domain, world, number, won, us, two, win, uk, british, list	music, servic, technolog, download, file, softwar, digit, firm, patent, content, network, play, offer, radio, control, compani, new, system, onlin, programm
net, user, broadband, secur, servic, softwar, onlin, viru, mail, uk, pc, microsoft, comput, technolog, access, program, internet, million, mani, connect	us, market, compani, firm, sale, report, share, bank, price, product, appl, growth, analyst, record, economi, month, china, busi
game, play, win, last, england, european, world, first, two, player, week, nation, unit, club, final, open, us, new, eu, match	mr, govern, parti, elect, labour, minist, plan, blair, law, new, spam, campaign, public, tori, tax, told, brown, issu, rule, gener

the representation is more accurate. Further, smaller corpora are more likely to be able to be modeled by fewer topics, and it is simply easier to interpret a DSN with fewer nodes. We see this when comparing the DSNs constructed on the BBC News data with those on the Reddit data. In Figure 4.4, we color each node by its most prominent topic. We see clear evidence of several clusters and can clearly see nodes that bridge multiple topics ("WTF", for example). As we increase the number of topics to 30 on the Reddit dataset, shown in Figure 4.5, the DSN becomes far less intelligible.

A two-dimensional representation of high-dimensional space is less likely to show high-dimensional clustering due to the amount of information lost in dimension reduction. We see this illustrated when moving from 5 to 50 topics on the BBC news articles in Figures 4.1 and 4.3, respectively. While

Table 4.3 List of 50 topics from BBC News data – First 25 Topics

tv, broadcast, set, satellit, content, programm, bbc, servic, viewer, us, show, channel, watch, data, network, definit, new, high, europ, line	metal, offer, slug, os, game, slice, magnet, side, fresh, five, trim, titl, fun, guard, visual, geniu, level, pile, slick	radio, spanish, station, hip, hop, listen, world, tremor, rap, millan, spain, argentina, artist, languag, music, track, web, latinohiphopradio, product, fan
old, lead, oil, wast, new, york, energi, marathon, edinburgh, london, radcliff, henson, crude, ga, agassi, import, suppli, end, full, winter	tv, broadcast, film, channel, televis, video, servic, programm, offer, digit, demand, sky, launch, watch, uk, set, box, screen, subscript, terrestri	film, best, award, star, show, includ, won, actor, prize, nomin, winner, first, director, oscar, top, perform, win, new, last, two
secur, softwar, user, program, viru, microsoft, window, pc, mail, virus, firm, comput, anti, spywar, infect, malici, releas, machin, version, xp	spam, mail, net, domain, scam, new, spammer, number, servic, name, premium, internet, messag, compani, inform, protect, address, junk, buy, sent	game, play, titl, onlin, world, player, hour, gamer, life, xbox, video, real, releas, mani, half, day, addict, two, spend, printer
car, electron, run, vehicl, race, four, drm, product, asimo, technolog, leg, disabl, sport, part, left, result, maker, quit, gear, hors	power, light, intel, laptop, silicon, laser, chip, virgin, compon, deaf, dr, materi, data, energi, manufactur, technolog, reduc, paniccia, current, made	cell, chip, browser, microsoft, market, processor, comput, technolog, offer, first, playstat, power, pc, opera, analyst, voic, soni, net, new, releas
attack, net, site, traffic, spam, data, mr, network, firm, bot, target, mani, crimin, websit, machin, spot, lyco, screensav, servic, user	pictur, imag, print, photo, high, produc, alreadi, photograph, hdtv, camera, home, digit, linux, us, mani, taken, street, screen, resolut, help	club, chelsea, arsen, leagu, manag, player, play, liverpool, footbal, season, team, ad, goal, boss, premiership, newcastl, manchest, told, cup, real
music, servic, network, download, file, peer, technolog, share, digit, industri, month, copyright, content, video, song, offer, new, advertis, per, napster	mobil, phone, handset, camera, technolog, network, gadget, messag, servic, data, number, multimedia, send, text, fi, oper, wi, video, system	copi, user, file, call, piraci, net, dvd, system, skype, network, firm, servic, free, industri, movi, technolog, voip, peer, new, hack
music, player, digit, devic, consum, media, market, technolog, ipod, portabl, mobil, new, appl, firm, video, design, drive, content, gadget	hunt, project, ban, music, servic, relay, group, orchestra, call, valv, concert, gameboyzz, old, video, bsl, languag, effect, us, cost, sound	mr, govern, elect, labour, parti, minist, blair, public, plan, tori, brown, tax, told, issu, howard, campaign, prime, claim, leader, polit
pc, hitachi, market, report, million, new, hewitt, china, australian, number, bsa, maker, expect, russia, world, third, germani, comput, local	patent, european, eu, law, softwar, direct, parliament, bill, compani, legal, us, invent, new, commiss, reject, comput, protect, open, council, minist	dvd, hd, game, format, technolog, high, next, ray, definit, film, standard, blu, gener, studio, soni, qualiti, disc, ibm, support, industri
search, googl, web, yahoo, site, mr, ask, engin, blog, jeev, inform, internet, technolog, servic, becom, definit, dvd, high, ad, news		

the 50 topics extracted (given in Tables 4.3 and 4.4) are nearly all useful, the resulting DSN would be difficult to interpret without labels.

When we consider Figure 4.3 with labels, we see that each of the 5 labels are nearly entirely grouped together in space, which indicates that there is likely some high-dimensional clustering. However, without knowing these labels beforehand, it would be nearly impossible to see this clustering by eye (which is the intended purpose of constructing a DSN). In contrast, with 5 topics, while it is unlikely that a user would be able to reconstruct the original five groups of documents without the labels, it is clear that there *is* clustering. Thus, we have found the greatest success on relatively few (≤10) topics.

Consistent with our observation that DSNs are less effective with a larger number of topics, we've found that as the number of topics increases, the

Table 4.4 List of 50 topics from BBC News data – Last 25 Topics

data, phish, million, found, survey, research, compani, mani, lab, comput, store, firm, month, cab, crime, lose, three, number	research, search, robot, player, engin, women, found, result, differ, user, report, net, inform, american, show, paid, million, accord, survey	broadband, servic, onlin, uk, bt, connect, net, access, million, internet, user, digit, speed, technolog, librari, home, fast, report, number, accord
card, id, new, cabir, seafar, rang, control, around, duti, technolog, pass, campaign, assault, set, call, passport, graphic, differ, safer, bluetooth	sun, grid, servic, anim, mr, shoot, hunt, comput, remot, pro, control, offer, custom, price, hour, glazer, rifl, shot, let, underwood	appl, inform, product, iptv, journalist, lawsuit, leak, report, week, secret, sourc, su, legal, reveal, blogger, three, us, trade, court, mail
comput, site, robot, trust, visual, world, system, websit, net, run, project, technolog, tsunami, address, donat, problem, help, hardwar, mail, special	gerrard, gm, fiat, turkey, simplifi, italian, narrow, parmalat, turkish, swamp, euro, morient, poppin, alfa, unilev, loss, isinbayeva, red, reach, drogba	win, england, wale, game, ireland, first, half, side, six, coach, play, team, nation, last, player, franc, two, final, rugbi, start
world, record, lift, second, olymp, indoor, speed, european, race, champion, holm, us, first, top, jump, men, women, tfc, compet, gold	system, file, develop, mr, technolog, firm, bittorr, uwb, help, share, version, ea, consum, creat, site, legal, softwar, network, onlin, control	compani, mr, us, china, countri, report, state, govern, group, market, foreign, india, presid, pay, call, oper, develop, new, world, million
lord, call, made, first, vodafon, uk, advic, launch, bypass, legal, nine, life, becom, goldsmith, million, cellnet, answer	show, technolog, gadget, digit, tv, game, home, buy, video, next, predict, shop, audio, definit, player, soni, new, sound, high	band, album, chart, number, new, day, singl, rock, top, us, week, follow, hit, singer, flight, air, two, union
economi, growth, rate, econom, sale, rise, price, market, bank, expect, increas, month, figur, decemb, record, quarter, last, fall	nintendo, soni, consol, ds, handheld, psp, game, sale, first, us, machin, releas, japan, sold, devic, million, launch, europ, market, expect	israel, minor, republ, ethnic, switzerland, blackpool, cypru, kerr, island, shortlist, faro, manchest, confer, kenni, brighton, hotel, parti, ham, host, bournemouth
ukip, kilroy, silk, firm, system, mail, box, parti, email, month, bet, wood, amount, receiv, offer, last, financi, robert, traffic	us, compani, firm, share, cost, deal, dollar, execut, airlin, yuko, stock, cut, mr, bank, profit, unit, news, ms, may, financi	comput, podcast, commodor, reddi, world, home, new, curri, radio, product, indian, brand, honda, audienc, possibl, listen, net, develop, mani
drug, carpent, fbi, mail, investig, viru, dope, recipi, attach, warn, agenc, internet, cont, public, comput, purport, jone, secur, contain, statement	play, open, match, final, set, first, roddick, cup, unit, seed, beat, second, round, game, ferguson, point, davi, win, two, break	rule, new, uk, mr, consum, problem, week, firm, govern, respons, without, court, job, servic, judg, last, remain, cut, seen, suggest
mac, comput, mini, pc, mr, appl, machin, map, mous, job, small, user, cost, new, softwar, hard, monitor, languag, help, design		

overall stress of the DSN increases. However, in coloring the edges based on their similarity, we're able to see where information has been lost in representing the $K-$dimensional data in two dimensions. Where there are longer red edges, we see that two similar documents were unable to be drawn close together, and thus their similarity was not able to be effectively shown in the DSN. This gives us immediate sense of *where* stress is occurring in the DSN and where clustering is being shown more (or less) effectively. Therefore, we can rely both on our stress value and the edge coloring to indicate whether our DSN is likely to be useful.

We have also found that DSNs are best suited to relatively small corpora. With fewer edges to scale, there is less overall stress on the network, and thus the representation is more accurate. Further, smaller corpora are more likely to be able to be modeled by fewer topics, and it is simply easier to interpret a

Table 4.5 List of 10 topics from Reddit comment data

mayweath, man, marufreza, amemb, edigitalplac, aff, watch, eye, hand, pacquiao, fight, door, day, thought, vs, turn, never, around, tag	war, us, dog, histori, countri, vote, govern, polit, actual, state, american, vietnam, mani, parti, youtub, watch, point, thank, question, mean
play, game, art, team, thank, skin, champion, titl, paint, love, player, free, buy, riot, feel, better, mean, lot, point, actual	point, mean, question, moral, actual, someon, seem, first, read, guy, reason, link, feel, interest, differ, give, better, life, argument, thought
f*ck, gold, kid, love, us, sh*t, never, guy, thank, mean, actual, day, lot, someon, better, give, alway, life, feel, mani	blood, asian, type, white, org, wiki, data, differ, point, en, amp, wikipedia, black, feel, articl, us, donat, actual, music, american
read, book, game, show, episod, movi, stori, first, watch, charact, end, love, actual, light, question, feel, great, lot, thank, seri	car, amp, wiki, day, money, cost, pay, lot, actual, start, buy, question, first, new, never, fit, pretti, job, hous, feel
music, listentothi, wiki, compos, automat, amp, bot, feedback, contact, question, moder, titl, commun, new, tag, link, artist, perform, song	f*ck, god, video, actual, mean, point, black, sh*t, watch, thank, life, call, someon, never, feel, differ, guy, white, give, us

DSN with fewer nodes. We see this when comparing the DSNs constructed on the BBC News data with those on the Reddit data. In Figure 4.4, we color each node by its most prominent topic. We see clear evidence of several clusters and can clearly see nodes that bridge multiple topics ("WTF", for example). As we increase the number of topics to 30 on the Reddit dataset, shown in Figure 4.5, the DSN becomes far less intelligible.

Because this work is very similar to the work of Lee et al. (2012), it is useful to draw a connection between our results and theirs. With suitably few documents and topics, such as the DSN in Figure 4.4, our DSN closely resembles their "Cluster Relation View", which depicts document clustering based on their LDA-extracted topics. Their application is structured to best support relatively few topics, which confirms our finding that low-dimensional representations of high-dimensional clustering are most suitable for few topics.

However, our work deviates from theirs in one key way: while they construct similar document similarity networks, they show a limited view of the extent to which document topic distributions differ from one another. They place a large amount of emphasis on the most prominent topic displayed in each document, displaying each document near documents with the same most prominent topic. This ignores the potential to discover clustering in documents that may have similar topic distributions but may contain more even mixtures between two or more topics.

Table 4.6 List of 30 topics from Reddit comment data

movi, film, watch, great, love, though, termin, end, actual, scene, f*ck, pretti, man, first, origin, seem, better, feel, lot, show	imag, imgur, photoshopbattl, automat, compos, link, amp, jpg, wiki, feedback, bot, googl, titl, contact, discuss, moder, perform, question, concern
question, first, watch, better, actual, amp, day, read, new, f*ck, wiki, point, lot, start, though, us, thank, music, never, feel	link, f*ck, thank, actual, point, better, first, dog, guy, black, mean, sh*t, pretti, never, lot, us, call, white, question, differ
mayweath, aff, amemb, marufreza, edigitalplac, pacquiao, vs, fight, floydmayweathermannypacquiaolivestream, watch, html, box, click, floyd, ppv, amp, showtim	wiki, question, new, thank, read, first, feel, great, link, amp, day, reason, point, actual, watch, better, start, f*ck, music, titl
car, money, pay, buy, job, save, month, hous, price, credit, lot, day, loan, spend, start, sell, deal, food, help, first	question, wiki, actual, first, read, watch, thank, reason, lot, someon, man, differ, end, day, better, car, find, mean, new
music, listentothi, wiki, commun, amp, compos, new, feedback, read, place, tag, artist, contact, automat, bot, share, reason, song, genr, question	man, eye, hand, thought, turn, door, around, first, never, day, face, us, feel, love, circl, two, prompt, start, head, stop
light, energi, question, photon, earth, moon, forc, thank, orbit, mass, org, may, wikipedia, power, wiki, particl, solar, enough, differ, space	love, thank, sexi, f*ck, nice, hot, wow, ass, beauti, ye, great, tit, bodi, perfect, pictur, amaz, damn, girl, gorgeou, pic
amp, wiki, fit, blood, rule, start, weight, question, day, new, type, eat, bodi, feel, muscl, lot, compos, probabl, exercis, automat	girl, f*ck, ask, op, school, rule, feel, tifu, tell, talk, never, happen, name, kid, let, stori, centiped, guy, sh*t, read
show, watch, thank, game, question, episod, us, better, actual, first, feel, war, love, great, lot, never, f*ck, titl, mani, live	cost, appl, power, batteri, product, watch, energi, price, actual, gener, lot, solar, point, money, system, compani, question, amp, mean, pay
lake, plu, scienc, two, f*ck, tyson, amp, great, guy, watch, sh*t, neil, glass, gif, video, point, imgur, cool, photo, man	watch, question, thank, first, listentothi, music, day, actual, wiki, read, point, mani, sh*t, made, us, better, never, reason, link, action
feel, thank, read, day, point, amp, old, man, better, question, actual, wiki, mani, call, guy, never, love, music, mean, f*ck	read, book, game, actual, play, first, never, lot, f*ck, differ, mean, probabl, end, happen, pretti, team, though, stori, day, joke
women, someon, men, guy, age, feel, rape, woman, child, parent, give, case, tell, old, doctor, girl, differ, walk, life, seem	point, question, actual, read, wiki, us, f*ck, someon, day, never, better, amp, tell, mean, thank, guy, feel, around, though, may
god, f*ck, watch, thank, gif, guy, stori, sh*t, rape, man, love, dream, feel, never, op, actual, link, read, thought, happen	mean, point, music, read, question, first, reason, actual, never, thank, wiki, listentothi, new, amp, find, around, mani, give, car, guy
moral, point, mean, argument, reason, believ, actual, interest, human, us, state, differ, question, law, seem, wrong, exist, give, philosophi, claim	wipe, god, toilet, hand, paper, sh*t, christian, stand, front, around, church, clean, water, natur, never, ball, word, mean, jesu, religion
kid, parent, polic, riot, child, life, anim, human, org, abus, mother, bird, mom, violenc, hit, chernobyl, white, wrong, son, radiat	amp, question, actual, first, f*ck, great, new, guy, music, watch, thank, mani, never, around, wiki, day, automat, though, alway, tell
amp, start, f*ck, question, wiki, thank, new, actual, feel, day, read, point, find, first, mean, better, love, lot, differ, guy	music, grooveshark, servic, spotifi, playlist, song, artist, f*ck, compani, record, pay, free, month, money, find, day, stream, amp, play, googl

Chapter 5

Conclusion

Through our work, we've successfully created a tool for visualizing text corpora such that evidence of clustering can be detected. Using python and the graphing application Gensim, we used Latent Dirichlet Allocation to extract topics from documents, then constructed a graph with nodes representing each document and edges with lengths drawn proportionally to the Hellinger distances between each document's topic distribution. Using edge coloring, we were able to find areas where our scaling was less effective, giving our model a built-in check of its own efficacy. Ultimately, we found that DSNs were most useful when constructed from relatively small corpora on 10 or fewer topics. Ultimately, we concluded that our DSN construction method could be highly useful when combined with similar visualization tools, such as those developed by (Lee et al., 2012), to aid future research in producing effective, meaningful cluster analysis on small text corpora.

Bibliography

Baker, Dylan. 2017. Dylan Baker's Thesis Code. URL https://github.com/dkbaker/thesis.

Bastian, Mathieu, Sebastien Heymann, Mathieu Jacomy, et al. 2009. Gephi: an open source software for exploring and manipulating networks. *ICWSM* 8:361–362.

Beel, Joeran, Bela Gipp, Stefan Langer, and Corinna Breitinger. 2016. Research-paper recommender systems: a literature survey. *International Journal on Digital Libraries* 17(4):305–338. doi:10.1007/s00799-015-0156-0. URL http://dx.doi.org/10.1007/s00799-015-0156-0.

Bird, Steven, Edward Loper, and Ewan Klein. 2009. *Natural Language Processing with Python*. O'Reilly Media Inc.

Blei, David M. 2012. Probabilistic Topic Models. *Communications of the ACM* 55(4).

Blei, David M., and John D. Lafferty. 2007. A Correlated Topic Model of Science. *The Annals of Applied Statistics* 1(1):17–35. URL http://repository.cmu.edu/cgi/viewcontent.cgi?article=2034&context=compsci.

Blei, David M., Andrew Y. Ng, and Michael I. Jordan. 2003. Latent Dirichlet Allocation. *Journal of Machine Learning Research* 3. URL http://www.jmlr.org/papers/volume3/blei03a/blei03a.pdf.

Chen, Liang, Yilun Wang, Qi Yu, Zibin Zheng, and Jian Wu. 2013. Wt-lda: user tagging augmented lda for web service clustering. In *International Conference on Service-Oriented Computing*, 162–176. Springer.

Deerwester, Scott, Susan T Dumais, George W Furnas, Thomas K Landauer, and Richard Harshman. 1990. Indexing by latent semantic analysis. *Journal of the American society for information science* 41(6):391.

Go, Alec, Richa Bhayani, and Lei Huang. 2009. Twitter sentiment classification using distant supervision. *CS224N Project Report, Stanford* 1:12.

Gong, Yihong, and Xin Liu. 2001. Generic text summarization using relevance measure and latent semantic analysis. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, 19–25. ACM.

Greene, Derek, and Pádraig Cunningham. 2006. Practical solutions to the problem of diagonal dominance in kernel document clustering. In *Proc. 23rd International Conference on Machine learning (ICML'06)*, 377–384. ACM Press.

Group, Algorithmics. 2009. Mdsj: Java library for multidimensional scaling. URL http://www.inf.uni-konstanz.de/algo/software/mdsj/.

Hagberg, Aric A., Daniel A. Schult, and Pieter J. Swart. 2008. Exploring network structure, dynamics, and function using NetworkX. In *Proceedings of the 7th Python in Science Conference (SciPy2008)*, 11–15. Pasadena, CA USA.

Ho, Ngoc-Diep. 2008. *Nonnegative Matrix Factorization Algorthms and Applications*. Ph.D. thesis, Université Catholique de Louvain.

Hofmann, Thomas. 1999. Probabilistic latent semantic indexing. In *Proceedings of the 22Nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 50–57. SIGIR '99, New York, NY, USA: ACM. doi:10.1145/312624.312649. URL http://doi.acm.org/10.1145/312624.312649.

Hunter, J. D. 2007. Matplotlib: A 2d graphics environment. *Computing In Science & Engineering* 9(3):90–95. doi:10.1109/MCSE.2007.55.

Jones, Eric, Travis Oliphant, Pearu Peterson, et al. 2001–. SciPy: Open source scientific tools for Python. URL http://www.scipy.org/.

Lee, Daniel D., and H. Sebastian Seung. 2001. Algorithms for non-negative matrix factorization. In *Advances in Neural Information Processing Systems 13*, eds. T. K. Leen, T. G. Dietterich, and V. Tresp, 556–562. MIT Press. URL http://papers.nips.cc/paper/1861-algorithms-for-non-negative-matrix-factorization.pdf.

Lee, Hanseung, Jaeyeon Kihm, Jaegul Choo, John Stasko, and Haesun Park. 2012. ivisclustering: An interactive visual document clustering via topic modeling. In *Computer Graphics Forum*, vol. 31, 1155–1164. Wiley Online Library.

Lee, Hyekyoung, Jiho Yoo, and Seungjin Choi. 2010. Semi-Supervised Nonnegative Matrix Factorization. *IEEE Signal Processing Letters* 17(1).

Lin, Chih-Jen. 2007. Projected gradient methods for nonnegative matrix factorization. *Neural Comput* 19(10):2756–2779. doi:10.1162/neco.2007.19.10.2756. URL http://dx.doi.org/10.1162/neco.2007.19.10.2756.

McKinney, Wes. 2010. Data structures for statistical computing in python. In *Proceedings of the 9th Python in Science Conference*, eds. Stéfan van der Walt and Jarrod Millman, 51 – 56.

Moghaddam, Samaneh, and Martin Ester. 2011. Ilda: Interdependent lda model for learning latent aspects and their ratings from online product reviews. In *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval*, 665–674. SIGIR '11, New York, NY, USA: ACM. doi:10.1145/2009916.2010006. URL http://doi.acm.org/10.1145/2009916.2010006.

Murphy, Kevin P. 2012. *Machine Learning: A Probabilistic Perspective*. Cambridge, Massachusetts: MIT Press.

Paatero, Pentti, and Unto Tapper. 1994. Positive matrix factorization: A non-negative factor model with optimal utilization of error estimates of data values. *Environmetrics* 5(2):111–126. doi:10.1002/env.3170050203. URL http://dx.doi.org/10.1002/env.3170050203.

Pak, Alexander, and Patrick Paroubek. 2010. *Twitter as a Corpus for Sentiment Analysis and Opinion Mining*. Ph.D. thesis, Universite de Paris-Sud, Orsay Cedex, France.

Pérez, Fernando, and Brian E. Granger. 2007. IPython: a system for interactive scientific computing. *Computing in Science and Engineering* 9(3):21–29. doi:10.1109/MCSE.2007.53. URL http://ipython.org.

Ramage, Daniel, Paul Heymann, Christopher D. Manning, and Hector Garcia-Molina. 2009. Clustering the tagged web. In *Proceedings of the Second ACM International Conference on Web Search and Data Mining*, 54–63. WSDM '09, New York, NY, USA: ACM. doi:10.1145/1498759.1498809. URL http://doi.acm.org/10.1145/1498759.1498809.

Reddit. 2015. May 2015 Reddit Comments. URL https://www.kaggle.com/reddit/reddit-comments-may-2015.

Řehůřek, Radim, and Petr Sojka. 2010. Software Framework for Topic Modelling with Large Corpora. In *Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*, 45–50. Valletta, Malta: ELRA. http://is.muni.cz/publication/884893/en.

Saha, Ankan, and Vikas Sindhwani. 2012. Learning evolving and emerging topics in social media: a dynamic nmf approach with temporal regularization. In *Proceedings of the fifth ACM international conference on Web search and data mining*, 693–702. ACM.

Slxu. 2009. Plate notation of the smoothed lda model. URL https://en.wikipedia.org/wiki/File:Smoothed_LDA.png.

Stevens, Keith, Philip Kegelmeyer, David Andrzejewski, and David Buttler. 2012. Exploting Topic Coherence over many models and many topics. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, 952–961. Jeju Island, Korea: Association for Computational Linguistics. URL http://aclweb.org/anthology/D/D12/D12-1087.pdf.

Wasserman, Stanley, and Katherine Faust. 2009. *Social network analysis: methods and applications*. Cambridge University Press.