Chapter 5

Probabilistic Text Reuse

Overview

Many collections of related documents contain passages of text that are highly similar. Discovering these passages can be an informative way to represent and understand interesting hidden structure in the document collection. In this paper, we present a new probabilistic model for document collections with repeated text passages. Our Probabilistic Text Reuse (PTR) model takes a generative approach and assumes that documents are composed of passages, where passages are either drawn from a canonical set of word sequences or *ideas*, or from a background language model. We model ideas as probabilistic finite state transducers, which generate each of the words in our sequence with some probability of matching, substitution, addition, or deletion. We also present algorithm for learning these ideas and their locations. Finally, we illustrate the utility of our model by finding common ideas in a large set of public comments on proposed U.S. net neutrality regulations and repeated sections of text in U.S. congressional bills. This paper provides a global objective function for text reuse, algorithms for optimizing this function, and an application for understanding a large collection of documents.

5.1 Introduction

Making sense of a large collection of text documents is an important task in many domains. For example, in the realm of politics and public policy, politicians may receive large numbers of letters from their constituents, and government agencies may receive millions of comments on proposed legislation. Citizens and journalists may be interested in the evolution of bills, regulations, and other documents released by government agencies or whistleblowers. Manually reading these large collections of documents may be impractical; automated and accurate ways of representing the content and viewpoints of these datasets are needed.

In each of the aforementioned tasks, the collections of text documents have a common characteristic: there are many instances of reused text. In some cases, such as when an advocacy organization encourages people to submit a form letter, entire documents can be repeated; in others, such as a collection of all bills introduced in a legislative body, smaller parts of documents may be reused. These duplicates can also be noisy, either due to technical processes (such as optical character recognition or text formatting variations) or human editing efforts (such as changes in wording or accidental copying mistakes or changes). Research in text reuse [e.g. 49, 56, 28] aims to understand documents through patterns of repeated or approximately repeated text.

In this chapter, we present a principled, probabilistic approach to capture and quantify these instances of text reuse in a corpus. We describe Probabilistic Text Reuse (PTR), a generative model for text reuse. Our PTR model explicitly learns a latent set of canonical text passages, which we term "ideas," that are directly useful to a person seeking to make sense of a large collection of documents. Our approach allows us to handle uncertainty and noise in text reuse — many documents may contain the same "idea" but with slight variations, and we can discover those examples in a principled manner. In this way, we take an important step towards automatically discovering common ideas from such large collections.

The contributions of our work are as follows:

1. We present Probabilistic Text Reuse (PTR), a generative model for making sense of collections of documents that have instances of repeated text. To the best of our knowledge, the PTR model is the first text reuse model with a global objective function, enabling patterns of text reuse to be evaluated in a principled, quantitative manner.

2. We present scalable algorithms for finding good parameter settings for PTR, including the discovery of ideas, partitions, and assignments.

3. We show the utility of PTR by illustrating how it is helpful for making sense of tens of thousands of citizen comments on the 2014 proposed rules by the Federal Communications Commission (FCC) related to net neutrality, particularly in comparison to more naive approaches to finding reused text and models that do not take into account word ordering. We also show how it finds repeated passages of text in bills introduced in Congress over a two-year period.

5.2 Related Work

5.2.1 Text Reuse Approaches

Our work is driven by interest, particularly among computational social scientists, in text reuse methods to understand large text corpora. Recent research involving text reuse includes mapping the diffusion of ideas in 19th-century newspapers during the U.S. Civil War [49], quantifying party contributions to the 2010 U.S. healthcare reform bill (Obamacare) [56], and tracing policy idea trajectories in Financial Crisis-related legislation [28]. In these domains, text reuse is a promising approach because there are substantial instances of copied or repeated text. As well, the content of the instances themselves are interpretable and often meaningful to social scientists. Determining whether text reuse exists, where it occurs in a collection of documents, and the content of repeated text are precisely the goals of both this literature and this current paper.

Currently, text reuse researchers typically use a chain of deterministic methods to find repeated sections of text in documents. In [49], for example, the researchers build a hash table of n-grams to find similar sections, run local sequence alignment algorithms to find matching pairs, and then use agglomerative clustering to group together similar passages. Each of these steps contains parameters and assumptions, such as n-gram window sizes, local sequence alignment costs, and clustering criteria, that could affect the quality of the discovered instances of text reuse. The gap that our work seeks to address is the lack of a global objective function for evaluating the quality of a text reuse solution.

5.2.2 Probabilistic Models of Text Corpora

Probabilistic generative models of text, in which words are drawn from some meaningful probability distribution over words, are a principled approach to modeling text corpora. They have at least three beneficial properties: 1) they assume a plausible, interpretable method by which the text data is generated; 2) they draw from a rich set of probability concepts and algorithms for inferring parameters; and 3) they have been shown to be empirically useful for language analysis tasks.

For instance, building a probabilistic model of text that is likely to appear in a collection of documents involves the following: 1) assuming that documents are drawn from a probability distribution over words; 2) inferring the probabilities of words by counting their relative frequencies, which can be seen as a form of maximum likelihood estimation; and 3) applying the model for prediction tasks such as speech recognition or machine translation, or visualizing the results s a word cloud, with more frequent words being more prominent.

A relevant latent-variable approach to characterizing text is probabilistic topic models, which assume that there exists a latent set of topics (multinomial distributions over words), and that each document is a multinomial distribution over topics. Popular methods for topic modeling include Latent Dirichlet Allocation (LDA) [7] and Probabilistic Latent Semantic Analysis (PLSA) [22]. Variants of these methods have been used to understand large text corpora ranging from research fields from scientific articles [6] to political agendas in U.S. Senate press releases [17]. The highest-weighted terms in a topic often provide some sense of the "meaning" of the topic, and the corresponding topic weights for a document can indicate what the document is about.

While they have value, these word-based approaches suffers from limitations for making sense of the ideas in large document collections. First, they treat documents as unordered bags of words, losing all of the language structure and voice that inscribe meaning to pieces of writing. Second, reading the highest-probability words in a topic is not always effective at interpreting a topic model's output, a challenge that has sparked additional work in labeling topics in topic models [33, 43, e.g.] and efforts to produce more human-interpretable topics [35]. In practice, for a human to make sense of these documents and topics, it might be necessary to post-topic modeling analysis and use the highly weighted words in topics as search terms to find and read relevant documents. While there are variants of LDA that move beyond unigrams into larger n-grams, they generally do not recover clauses or longer phrases [e.g. 54, 55, 31]. For these reasons, the outputs of probabilistic topic models alone may be insufficient for making sense of the content of large document collections.

5.2.3 Text Summarization

Our task is distinct from, though arguably related to, the task of automatic text summarization, particularly in the multi-document setting [see, e.g., 38, 13, 16]. Finding frequently repeated sections of text in a large collection of documents might be useful for understanding the contents of the corpus, but we do not focus on building a written, coherent summary in this work. Identifying patterns of repeated text may itself be useful for understanding the corpus, or the repeated passages may be useful for other downstream tasks.

5.3 Probabilistic Text Reuse Model

We take a generative modeling approach and assume that our documents are comprised of text passages, which are sequences of words of varying-length "ideas" that repeat throughout the corpus. Our working hypothesis in this Probabilistic Text Reuse (PTR) model is that these ideas are useful units, both for quantitatively explaining the documents and for humans to make sense of them.

Let D be the set of N text documents, $d_1, d_2, \ldots, d_N$. Each document, d_n, consists of T_n words, $w_1, w_2, \ldots, w_{T_n}$; let V be the set of unique words, meaning that $|V|$ is the size of the vocabulary. PTR places the following generative process on the document collection D:

1. We assume that there exist I "text sequence generators" or *ideas* $\{k_1, \ldots, k_I\}$. These ideas can be of varying length. As described further below in Section 5.3, ideas are modeled as probabilistic finite state transducers (PFSTs), which are probabilistic functions over sequences of words.

2. A document d_n is a sequence of *partitions*, which contain passages of text. Each partition is either generated from a probabilistic finite state transducer (PFST) of idea k_i or from a background language model. We use z_{nm} to denote the start index of the m^{th} partition in document and a_{nm} to denote the idea or background model associated with the m^{th} partition.

In contrast to models that assume that documents are unordered bags of words, the ideas k_i consist of ordered sequences of words, which can be of varying length (generally, as shown below, ideas can be as short as clauses to paragraphs). Thus, our PTR model can capture more interpretable, meaningful entities than a standard topic model, which usually describes documents as unordered mixtures of n-grams.

Let K be the collection of ideas $\{k_i\}$, Z be the collection of partition indices $\{z_{nm}\}$, and A be the collection of assignments $\{a_{nm}\}$. We place priors $P(K)$, $P(Z)$, $P(A)$ over each of these collections; given the data D, the joint distribution $P(D, K, Z, A)$

is given by

$$\begin{aligned} P(D,K,Z,A) &= P(K,Z,A) \cdot P(D \mid K,Z,A) \\ &= P(K) \cdot P(Z) \cdot P(A) \prod_{\text{partitions},\, z_{nm}} P(d_{z_{nm}} \mid k_{a_{nm}}) \end{aligned} \tag{5.1}$$

where we use $d_{z_{nm}}$ to denote the text associated with the partition z_{nm} and the assignment a_{nm} can be either an idea k_i or a background language model, which we denote by k_0.

Our objective is to find the parameter settings for K, Z, and A that maximizes the objective function, i.e.:

$$\arg\max_{K,Z,A} P(D,K,Z,A) = \arg\max_{K,Z,A} P(K,Z,A) \cdot P(D \mid K,Z,A) \tag{5.2}$$

We describe each of these factors below.

Idea Model, $Pr(K)$ In our model, we consider *frequently reused passages of text* as the ideas: Through some societal process, these ideas occur in multiple documents in our collection. We first generate the length of an idea L_i from a uniform distribution between $N_{k,min}$ and $N_{k,max}$ words and then generate each of the words a unigram language model:

$$\begin{aligned} L_i &\sim \text{Unif}(N_{min}, N_{max}) \\ k_i(l) &\sim \pi(w) \\ P(K) &= \prod_{i=0}^{I} \frac{1}{N_{max} - N_{min}} \cdot \prod_{\text{words}, w_l \in k_i} \pi(w_l) \end{aligned}$$

where $\pi(w)$ is the probability of generating word w; we set $\pi(w)$ to be the empirical probabilities of each word in the corpus. Thus, the cost of generating an idea is the probability of generating its length and each of its words from the background language model. More frequent words in the corpus tend to have higher probabilities, so they are more likely to be part of an idea.

We seek to discover ideas that are longer phrases, as opposed to short n-grams that may occur frequently but not be as meaningful. Consequently, we set N_{min} to be about a few words long (we use $N_{min} = 5$ in our experiments. It is also worth noting that our model might consider "boilerplate" text (e.g. addresses, form language, or preambles in legal documents) as "ideas" — the definition of "boilerplate" might differ across applications. However, in practice, these boilerplate ideas should be easily identifiable from the output of the model and could be appropriately disregarded by a human analyst.

Partitions Model, $P(Z)$ We posit that the number of partitions in a document $|z_n|$ is drawn from a uniform distribution $\text{Unif}(0, N_z)$. We choose to model $|z_n|$, rather than the specific locations z_{nm}, for robustness — each partition can independently choose its length. As a result, the probability of a set of partitions $P(Z)$ is simply

$$P(Z) = (\frac{1}{N_z})^N$$

$$\begin{aligned} \text{partitions per document} &\sim \text{Unif}(1, N_z) \\ P(Z) &= \prod_{\text{documents},n}^{N} (\frac{1}{N_z - 1}) \\ &= (\frac{1}{N_z - 1})^N \end{aligned}$$

Assignments Model, $P(A)$ Each partition is assigned either to an idea k_i or the background language model with some hidden parameters θ_i:

$$P(A) = \prod_{n,m} \prod_{i=0}^{I} \theta^{\mathbb{I}(a_{nm}=i)} \tag{5.3}$$

Text Passage Model, $P(D|Z, K, A)$ A passage, d_i, is either generated from the background language model k_0 or from an idea, k_i. In this work, we simply use a unigram background language model:

$$P(d_{z_{nm}} \mid a_{nm} = k_0) = \prod_{l=1}^{L_{nm}} \pi(d_{z_{nm}}(l)) \tag{5.4}$$

where $\pi(w)$ is the empirical frequency of word w is the corpus.

Since both the partition text d_z and the assigned idea k_a are sequences of words, we need to define a conditional probability distribution of any sequence of words given an idea. Conceptually, we seek a function that does the following:

$$P(d_z \mid k_a) = \text{stochastic edit distance between } d_z \text{ and } k_a \tag{5.5}$$

We define this sequence using finite state transducers (FSTs). FSTs are finite automata that read an "input" string and, in addition, produce an "output" string. In PTR, we use probabilistic finite state transducers (PFSTs) for our ideas, which have valid probabilities as weights in each of the transitions and place a valid distribution over possible output strings. Specifically, we use the PFST formulation described in [14]. The PFST for each idea k_i is defined as follows:

- The states of the PFST are the start state q_0, states $q_1, \ldots, q_{L_i}$ corresponding to each word $w_1, \ldots, w_{L_i}$ that make up the idea k_i, and an end state q_f.

- For each of the states from q_0 to q_{L_i}, there are $|V|$ self-transitions, corresponding to insertions of any of the $|V|$ words in th vocabulary, and $|V| + 1$ transitions to the next state. The transitions are described with two symbols: one for the input string and one for the output string that are, by convention, represented with a colon (:) separation. In the case of an idea currently in the state corresponding to the word “a”, the possible $2|V|+1$ transitions can be represented as follows:

  - $a : a$ advances the FST to the next state with a match probability p_m.
  - $a : b$ advances the FST to the next state with a substitution ($p = p_s/|V|$ for each of $|V|-1$ words in the vocabulary). There are $|V|-1$ possible substitutions.
  - $a : \epsilon$ advances the FST to the next state with a deletion probability p_d.
  - $\epsilon : b$ keeps the FST in the current state, and represents an insertion with probability p_i. There are $|V|$ possible insertions, corresponding to each of the words in the vocabulary.

- The state corresponding to the last word, q_{L_i}, has just two types of transitions: with probability p_i, the PFST stays in the current state, representing an insertion (with probability p_i). Otherwise, the string terminates with probability $1-p_i$.

With these operations, it is possible to generate any string from the PFST. In fact, in general, there are multiple ways to generate an output string from a PFST. We can efficiently compute the probability of an observed string—which involves summing over the probabilities of all of the paths in the PFST—through dynamic programming, i.e. the forward algorithm. For this work, we fix the parameters $\{p_m, p_s, p_d, p_i\} = \{0.8, 0.1, 0.1/|V|, 0.1\}$, which provides high weights for similar or identical strings and equal penalties for an addition, substitution, or deletion. The corresponding language model for a three-word idea is shown in Figure 5-1.

For any string, we can infer the probability that it was generated by the idea, along with most likely sequence of matches, additions, deletions, and substitutions, using a dynamic programming algorithm.

To summarize, Figure 5-2 shows the purpose of PTR: The goal is to take a large

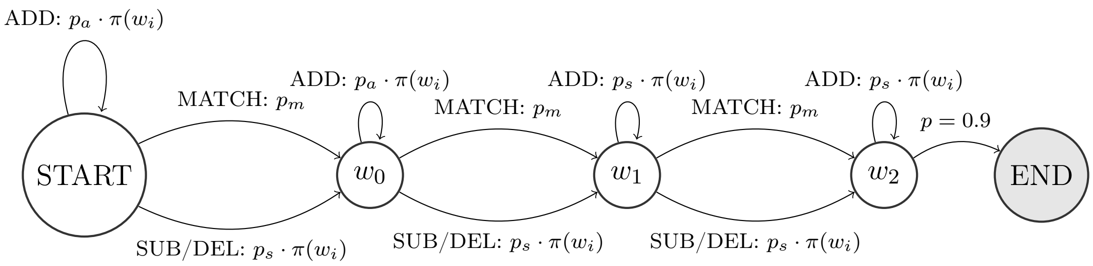
ADD: $p_a \cdot \pi(w_i)$
START
MATCH: p_m
SUB/DEL: $p_s \cdot \pi(w_i)$
ADD: $p_a \cdot \pi(w_i)$
w_0
MATCH: p_m
SUB/DEL: $p_s \cdot \pi(w_i)$
ADD: $p_s \cdot \pi(w_i)$
w_1
MATCH: p_m
SUB/DEL: $p_s \cdot \pi(w_i)$
ADD: $p_s \cdot \pi(w_i)$
w_2
$p = 0.9$
END

Figure 5-1: Probabilistic finite state transducer (PFST) for three-word idea

collection of unlabeled documents (D) and infer both the ideas (K) for the corpus and the partitions (Z) and assignments (A) for each document.

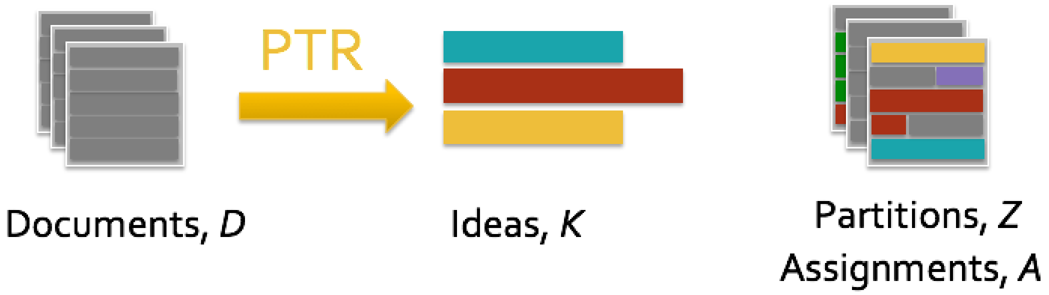

Figure 5-2: Inputs and outputs of Probabilistic Text Reuse (PTR)

5.4 Inference

Inference in our Probabilistic Text Reuse model involves inferring the ideas K, the partitions Z, and the assignment of partitions to ideas A. To obtain an approximate MAP solution to the objective in equation 5.1, we iteratively optimize the each of these three sets of hidden variables given the rest.

5.4.1 Initialization

A good initialization of the ideas k can greatly reduce the number of iterations required to converge to the approximate MAP solution. We first find a large number of word sequences that may be recurring ideas in the dataset. Through a hash table, we can count and mark the locations, which we call "anchors", of the most common n-grams of size 5, the minimum length of an idea in our model. We choose a random subset of these anchors and perform local sequence alignment on the other anchor positions to find a common idea string. An alternative, more scalable approach is to run a sentence boundary detector on the data, and to consider sentence boundaries as passage boundaries. In a later step, we can then merge together ideas that are more than one sentence in length.

5.4.2 Updating Ideas

Given a set of partitions assigned to a particular idea k, updating the associated idea string is a form of the Steiner consensus string problem [19]. In the general case, finding the Steiner consensus string (the idea string that would minimize the distance to all of the strings in the set) is NP-hard [47]; however, choosing the best-performing representative from the set of partitions as the idea is guaranteed to be a reasonable approximation — the best performing string in the set will be no worse than twice the true optimum. We update the idea by choosing the string among those assigned to the idea that maximizes the probability of the set of partitions currently assigned to the idea.

5.4.3 Updating Partitions and Assignments

Next, we map sequences of text to our ideas or the background language model. Using dynamic programming, we find the probability that the text was generated from the PFST corresponding to each of the ideas. If this probability is greater than the the passage's score from the background language model, then this passage is assigned to the idea. Each idea now has a set of passages assigned to it that either match exactly or approximately. We devise a novel dynamic programming algorithm to accomplish this task — the alignment of an idea to a substring of a document is independent of the preceding and following substrings. As a result, it is possible to compute all possible alignments and then find the most probable path through the document.

5.4.4 Merging Ideas

Merging similar ideas means that the cost of creating one of the ideas is no longer needed, and it also increases the $P(A)$ assignment term. However, merging dissimilar ideas can reduce the likelihood, because having more ideas can match passages of text more closely. We propose the following approach to merge ideas:

1. Count pairs of consecutive idea assignments across the corpus, where "consecutive idea assignment" means that an adjoining pair of partitions are assigned to two ideas.

2. Choose the most popular consecutive assignment and propose a new idea that consists of the merged versions of the two ideas. We also retain the two original ideas, in case there are other partitions throughout the corpus that most closely align with just one of these ideas.

3. If the likelihood of the partitions belonging to these consecutive ideas increases when they are assigned to a single, merged idea, then the ideas are merged.

5.4.5 Assignment Probabilities

During inference, we fit the maximum likelihood parameters to θ_i, which results in

$$\Pr(A) = \prod_{n,m} \prod_{i=0}^{I} \frac{N_i}{|A|}^{\mathbb{I}(a_{nm}=i)} \tag{5.6}$$

where N_i is the number of passages assigned to idea i and $|A|$ is the total number of passages.

5.5 Dataset: FCC Comments on Net Neutrality

We use submissions from the first comment period (May 15 to July 15, 2014) to the U.S. Federal Communications Commission (FCC) on its proposed rules on "Protecting and Promoting the Open Internet". This collection was the FCC's largest public comment collection to date and is publicly available. This comment period triggered enormous reaction from citizens and civil-society groups, including citizens and civil-society groups advocating for network neutrality. For example, many comments encouraged the FCC to classify and ISPs as “common carriers” (similar to telephone companies), which would, like telephone companies, empower the FCC to enforce network neutrality. Other comments, meanwhile, urged the FCC and the government not to get involved in regulations that could stifle innovation.

In general, many of comments are relatively short, often just a single sentence or paragraph. They also contain large numbers of form letters, some of which have customizable passages, prepared by civil-society groups, who mobilized individuals to submit them. We sample approximately 80,000 comments for our training set. Table 5.1 provides summary statistics of the corpus that we use to analyze our dataset.

Table 5.1: Summary of FCC comment corpus (N=800000)

mean # of words per comment (mean and standard deviation)	131 ± 2681
# of unique comments	650,300

Making sense of this large dataset of comments is a challenge for regulators and other interested parties that we seek to solve. We compare PTR with other methods in the next section.

Model Parameters We set the minimum and maximum idea lengths to be $N_{k,min} = 5$ and $N_{k,max} = 45$, and the maximum number of partitions in a document to be $N_z = 50$. These values were chosen to ensure that ideas tend to be more like sentences or sequences of sentences than single words or common multiword expressions. The PFST parameters were set to $p_m = 0.8$, $p_d = 0.1$, $p_i = 0.1/|V|$. These values were chosen to provide high weights for similar or identical strings and equal penalties for an addition, substitution, or deletion. As we describe in our discussion, future work could involve learning these parameters, perhaps on an idea-specific basis or in a tied manner.

5.6 Results

PTR finds passages of varying length and gives more accurate counts. In Table 5.2 and Table 5.3, we show an idea discovered by the sentence-boundary method with higher counts (due to finding approximate matches).

5.6.1 Noteworthy Top Ideas

Many of the top ideas come from templates from civil society groups, which encouraged people to submit form letters that they had prepared. Some examples of these ideas are below:

CREDO Action: The corpus contains 93,711 comments that include the following text: *"As an Internet user who believes strongly in the importance of a free and open Internet, I urge the FCC to reclassify broadband Internet access as a telecommunications service, and save Net Neutrality. In addition, the FCC should reject the proposed rules that would allow Internet service providers to divide the Internet into fast lanes for wealthy corporations and slow lanes for the rest of us."*[1]

Fight for the Future: 89,989 comments contain the following text: *"Net neutrality is the First Amendment of the Internet, the principle that Internet service providers (ISPs) treat all data equally. As an Internet user, net neutrality is vitally important to me. The FCC should use its Title II authority to protect it. Most Americans have only one choice for truly high speed Internet: their local cable company. This is a political failure, and it is an embarrassment. America deserves competition and choice..."*.[2]

Electronic Frontier Foundation: Approximately 68,000 comments begin with the following text: *Dear FCC, My name is Steve Roberts and I live in West Lafayette, IN. Net neutrality, the principle that Internet service providers (ISPs) treat all data that travels over their networks equally, is important to me because without it..."*.[3]

[1] See "CREDO Action: URGENT: Tell the FCC: Don't Kill The Internet", http://act.credoaction.com/sign/fcc_nn_comments_2014.

[2] See "Fight for the Future", https://www.fightforthefuture.org/.

[3] See "Electronic Frontier Foundation: Net Neutrality", https://www.eff.org/issues/net-neutrality.

American Commitment: Over 9,300 comments calling for less government intervention in Internet regulation (a view opposite to the ones expressed above) contained the following text: *"As an American citizen, I wanted to voice my opposition to the FCC's crippling new regulations that would put federal bureaucrats in charge of internet freedom, and urge you to stop these regulations before they're enacted…Please stop the FCC's dangerous new regulations, and protect the future of internet freedom here in America."*[4]

5.6.2 Less-Common Voices

PTR is able to discover variations of less-common voices — it aggregates passages that share a significant amount of common text. Tables 5.3 and 5.2 show examples of relatively rare ideas that, as a result of PTR's ability to capture variation, are included in the set of ideas that characterize the FCC net neutrality comments.

Table 5.2: Variations on "the internet should be open" (168 assignments)

variation on idea (sample of 21 variations)
the internet should be **publicly owned**
the internet should **never be regulated**
the internet should be **divided**
the internet should be open **and neutral**
the internet should be open **to everyone equally**
the internet should be **an** open **platform**
the internet should be **equal opportunity**
the internet should be **taxed**
the internet should be **fair**
the internet should **absolutely** be open

5.6.3 Baseline Comparisons

Topic Modeling: Table 5.4 shows the five words with the highest probability in selected topics of a 100-topic model using Latent Dirichlet Allocation [7]. The topics

[4] See "American Commitment: 808,363 Americans Tell the FCC: 'Do Not Regulate the Internet,"' `https://www.americancommitment.org/content/do-not-regulate-internet`.

Table 5.3: Variations on "keep the internet a level playing field" (244 comments)

variation on idea (sample of 90 variations)
keep the internet **an even** playing field
keep the internet **an open** playing field
keep **it** a level playing field
keep **net neutrality keep** a level playing field
keep the **net** a level playing field
keep the internet a level playing field **that it is**
keep the internet a **fair** playing field
keep the **net on** a level playing field
keep the internet **open as a fair** playing field
keep the **media landscape** a level playing field

illustrate some general themes that emerge from the corpus; however, it is difficult to directly discern submitted viewpoints from the LDA results themselves.

Table 5.4: Top ten words from selected topics of LDA model with 50 topics

topic	**keywords**
1	isps, use, slow, able, destroy
2	isps, important, new, services, better
3	communications, corporations, rules, reclassify, federal
4	free, access, equal, corporations, open
5	common, carriers, reclassify, carrier, broadband
6	companies, small, businesses, business, innovation
7	people, corporations, right, corporate, government
8	pay, content, companies, access, speed
9	wealthy, save, user, addition, believes
10	comcast, like, verizon, cable, time

Figure 5-3 shows the results of clustering documents by topic distributions. We applied k-means clustering on the document-topic distributions, then sized the nodes in the displayed plot by the number of documents in those clusters. The layout of the graph is based on a force-equilibrium approach, in which the attraction between two clusters is proportional to their similarity and a pair of clusters has a visible edge if at least one of the clusters is among the top-five closest clusters of the other. This tends to put larger and more central nodes closer to the center of the layout. As the

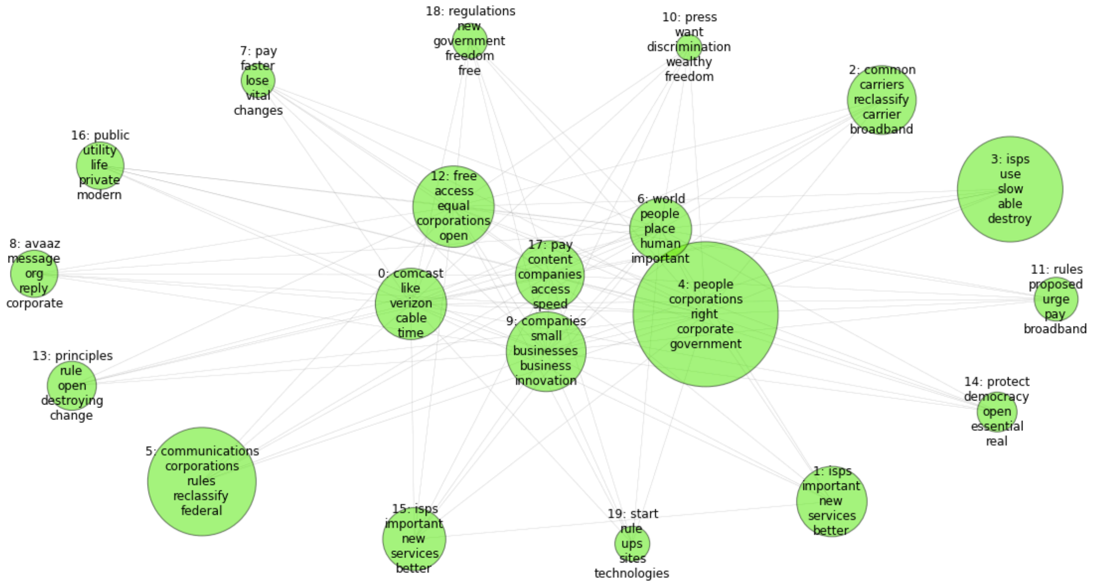
18: regulations new government freedom free
10: press want discrimination wealthy freedom
7: pay faster lose vital changes
2: common carriers reclassify carrier broadband
16: public utility life private modern
3: isps use slow able destroy
12: free access equal corporations open
6: world people place human important
8: avaaz message org reply corporate
17: pay content companies access speed
0: comcast like verizon cable time
4: people corporations right corporate government
11: rules proposed urge pay broadband
9: companies small businesses business innovation
13: principles rule open destroying change
14: protect democracy open essential real
5: communications corporations rules reclassify federal
1: isps important new services better
15: isps important new services better
19: start rule ups sites technologies

Figure 5-3: Topic-based clusters of public comments, with nodes sized by the number of comments in the cluster

figure shows, topic models can provide a useful global overview of the corpus, but they fail to capture the context in which words are used.

Common Sentences: A second approach is to run a sentence boundary detector and extract the most common sentences. As shown in Table 5.5, this process, by definition, results in coherent sentences. However, it turns out that the most frequent sentences are all from the most frequently occurring repeated comment.

5.6.4 Quantitative Comparison to LDA

One approach to quantitatively analyzing the results is to compute the likelihood of the data with respect to the model. Table 5.6 compares the log-likelihood/token for a baseline unigram language model, PTR, and a 50-topic LDA model trained on the data. Most notably, for the FCC corpus, PTR outperforms LDA on the dataset in terms of likelihood, which occurs because there is substantial text reuse — about 39% of the partitions are assigned to an idea. In contrast, while text reuse does occur in

Table 5.5: Top sentences in corpus, by frequency.

count	**sentence**
116,923	in addition, the fcc should reject the proposed rules that would allow internet service providers to divide the internet into fast lanes for wealthy corporations and slow lanes for the rest of us.
113,702	14-28 comments as an internet user who believes strongly in the importance of a free and open internet, i urge the fcc to reclassify broadband internet access as a telecommunications service, and save net neutrality.
111,240	title ii is the strong, legally sound way to enforce net neutrality.
111,225	this is the same trick they pulled last time.
111,225	isps are opposing title ii so that they can destroy the fcc's net neutrality rules in court.
111,221	the fcc should use its title ii authority to protect it.
111,221	please, let's not be fooled again.
111,211	it also causes tremendous economic harm.
111,211	that kills choice, diversity, and quality.
111,206	without net neutrality, a bad situation gets even worse.

the Congressional bills dataset, it represents a much smaller proportion of the corpus (approximately 1%). As a result, while the instances of text reuse are qualitatively interesting, LDA performs better at characterizing the dataset.

Table 5.6: Log Likelihood per token with unigram, LDA, and PTR models

model	**log-likelihood/token**
baseline unigram	-7.32
100-topic LDA	-6.48
PTR	-3.26

5.7 Discussion and Further Work

The results from PTR are better suited for making sense of this corpus than LDA or our other baseline methods. Specifically, it has the following advantages:

1. The outputs are more human-interpretable than the bags of words of other

topic modeling approaches. Our generative model is able to discover examples of text reuse throughout the corpus, cluster together similar passages of text as originating from the same idea, and find a reasonable exemplar (the "idea") that, in itself, is useful to read.

2. The model handles stop words, which, in practice, need to be removed from bag-of-words representations. In fact, stop words are an important component of PTR's representation of the data: Without them, it would be more difficult, if not impossible, to understand the passages of reused text in the comments.

3. Topic modeling does not necessarily produce topics that correspond to themes or ideas in the corpus. In LDA, for example, documents are multinomial distributions over topics, and topics are multinomial distributions over words. A topic in which few words have a relatively high amount of the probability mass could explain the data well, but it may not yield any practical significance for someone trying to make sense of a large corpus of documents. For example, topic 6 in Table 5.4 appears to be a set of words ("united", "states", "people", "fcc", etc.) that might appear very frequently throughout the corpus (but arguably should not be excluded as stop words). Similarly, while topics 1 and 3 are clearly about cable companies and Internet service providers (ISPs), the list of words poorly conveys the meaning of these statements.

4. In contrast, the ideas in the PTR model are more closely aligned to the FCC's goal of understanding the comments than the output of LDA. Finding passages of text that have high conditional probability for any of the topics is certainly possible, but it would require an additional, post-hoc analysis step. In contrast, PTR directly outputs useful-to-read ideas.

5. By assuming that passages of text are generated, with noise, from the set K, PTR identifies and finds similar statements. As a result, it provides more accurate counts or proportions of ideas than simply counting sentences, and is a more principled approach than simply clustering similar sentences afterward.

Figure 5-4 provides an illustration of how PTR can be applied to a text corpus to find key phrases: We can go directly from sequences of words (documents) directly to these PTR idea sequences. In contrast, models such as PLSA and LDA require an intermediate representation of documents, namely an unordered bag of words, which, especially in text corpora with substantial text reuse, results in a mismatch between the model and the data. In order to discover key phrases that occur in topic clusters, one needs to return to the original documents and somehow extract these key phrases. While PLSA and LDA have great utility for a general global understanding of a text corpus or for other applications, they are ill-suited for identifying commonly repeated phrases.

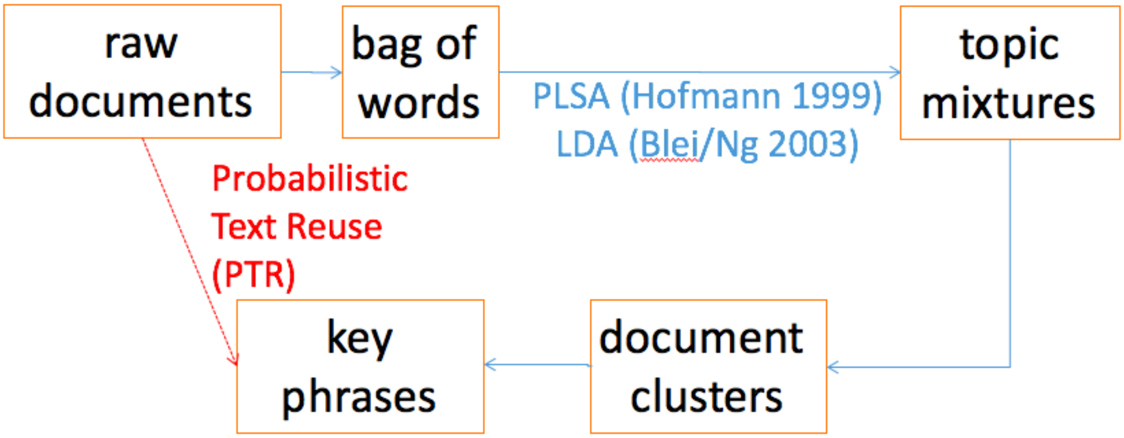

Figure 5-4: Summary of pipeline for finding key phrases with PTR vs. bag-of-words PLSA and LDA models

Future work on PTR could focus on efficiently finding the optimal set of partitions, ideas, and assignments, as well as more sophisticated background language models. More broadly, there are many interesting directions in learning the PFST parameters to go beyond text reuse and encode semantic similarity—which would be the eventual goal when extracting key ideas from large collections of text.

5.8 Conclusions

In this chapter, we introduce Probabilistic Text Reuse (PTR), a model that discovers and quantifies repeated passages of text in a large corpus of documents. On the FCC net neutrality comment dataset, we show that PTR qualitatively outperforms LDA and other baseline methods at presenting ideas from the corpus. Our approach could be useful for understanding key ideas of other large datasets, both in computational social science and in other domains.

Given the widespread prevalence of text reuse in legal and political documents, PTR could be applicable in many other large text collections. For measuring political speech, other public comment datasets collected by governments or online platforms or communities such as Facebook, Twitter, or Reddit might be useful. One could speculate that a "PTR web crawler" could find meaningful instances of significant text reuse on the Internet, perhaps bringing together online conversations or even automatically starting petitions or larger movements. More generally, these online platforms or Internet researchers might use text reuse more generally to examine how reused text propagates in their social and document networks, revealing the dynamics of how memes and other ideas spread. Meanwhile, text reuse detection could help public or private organizations break data silos (by discovering that similar text or documents are used in different parts of the organization), find new opportunities to collaborate, or even improve worker productivity, either by avoiding the duplication of the same manual work on a given document or passage of text (e.g. in language translation, customer service, or document approvals) or as an informative feature for machine learning systems. Returning to the government domain, text reuse occurs widely in bills and laws, and PTR could be useful for seeing the evolution of documents in institutions such as Congress (similar to the work in Chapter 4) or even across legislative bodies. It would be fascinating and informative, for example, to map out the diffusion of policy ideas across U.S. state lines by detecting text reuse in the bills introduced in all 50 states; going further, the influence of lobby groups (some of which are known to draft template bills) and other civil society movements might be measured with PTR. Ultimately, written language is highly rich and flexible, allowing people to express ideas in an infinite number of ways; empirically, though, text reuse occurs in a wide range of contexts, and detecting its existence, as illustrated in this chapter, can reveal important insights into large document collections.

Bibliography

[1] The Annotated 8 Principles of Open Government Data. http://opengovdata.org/. Accessed: 2014-05-31.

[2] Virginia Decoded. http://vacode.org/. Accessed: 2014-05-31.

[3] Tracing Policy Ideas From Lobbyists Through State Legislatures. http://sunlightfoundation.com/tools/churnalism-us//, 2013. Accessed: 2015-12-20.

[4] E. Scott Adler and John Wilkerson. Congressional Bills Project: 1973-2014. http://congressionalbills.org/, 2014.

[5] Jaime Arguello, Jamie Callan, and Stuart Shulman. Recognizing citations in public comments. *Journal of Information Technology & Politics*, 5(1):49–71, 2008.

[6] David M Blei and John D Lafferty. A correlated topic model of science. *The Annals of Applied Statistics*, pages 17–35, 2007.

[7] David M Blei, Andrew Y Ng, and Michael I Jordan. Latent Dirichlet Allocation. *the Journal of Machine Learning Research*, 3:993–1022, 2003.

[8] Thomas R. Bruce. Cornell Legal Information Institute. http://www.law.cornell.edu/, 2015. Accessed: 2014-05-28.

[9] Thomas R Bruce and Peter W Martin. The Legal Information Institute: What Is It and Why Is It. *Cornell Law Forum*, 20, 1994.

[10] Matthew Burgess, Eugenia Giraudy, Julian Katz-Samuels, Lauren Haynes, and Joe Walsh. Tracing Policy Ideas From Lobbyists Through State Legislatures. http://dssg.uchicago.edu/project/tracing-policy-ideas-from-lobbyists-through-state-legislatures/, 2015.

[11] Claire Cardie, Noah Smith, Anne Washington, and John Wilkerson. NLP Unshared Task in PoliInformats 2014. https://sites.google.com/site/unsharedtask2014/. Accessed: 2014-05-31.

[12] Sophie Chou, William Li, and Ramesh Sridharan. Democratizing Data Science. In *Data for Good: KDD at Bloomberg*, 2014.

[13] Janara Christensen, Stephen Soderland Mausam, Stephen Soderland, and Oren Etzioni. Towards Coherent Multi-Document Summarization. In *HLT-NAACL*, pages 1163–1173, 2013.

[14] Ryan Cotterell, Nanyun Peng, and Jason Eisner. Stochastic Contextual Edit Distance and Probabilistic FSTs. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics, ACL 2014, June 22-27, 2014, Baltimore, MD, USA, Volume 2: Short Papers*, pages 625–630, 2014. URL `http://aclweb.org/anthology/P/P14/P14-2102.pdf`.

[15] Sharon S Dawes. Stewardship and usefulness: Policy principles for information-based transparency. *Government Information Quarterly*, 27(4):377–383, 2010.

[16] Jade Goldstein, Vibhu Mittal, Jaime Carbonell, and Mark Kantrowitz. Multi-document summarization by sentence extraction. In *Proceedings of the 2000 NAACL-ANLPWorkshop on Automatic summarization-Volume 4*, pages 40–48. Association for Computational Linguistics, 2000.

[17] Justin Grimmer. A Bayesian hierarchical topic model for political texts: Measuring expressed agendas in Senate press releases. *Political Analysis*, 18(1):1–35, 2010.

[18] Justin Gross, Brice Acree, Yanchuan Sim, and Noah A Smith. Testing the Etch-a-Sketch Hypothesis: A Computational Analysis of Mitt Romney's Ideological Makeover During the 2012 Primary vs. General Elections. In *APSA 2013 Annual Meeting Paper*, 2013.

[19] Dan Gusfield. *Algorithms on strings, trees and sequences: computer science and computational biology*. Cambridge university press, 1997.

[20] Alexander Hertel-Fernandez. Who Passes Business's "Model Bills"? Policy Capacity and Corporate Influence in US State Politics. *Perspectives on Politics*, 12 (03):582–602, 2014.

[21] Alexander Hertel-Fernandez and Konstantin Kashin. Capturing Business Power Across the States with Text Reuse. In *Annual conference of the Midwest Political Science Association, Chicago, April*, pages 16–19, 2015.

[22] Thomas Hofmann. Probabilistic Latent Semantic Indexing. In *Proceedings of the 22Nd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '99, pages 50–57, New York, NY, USA, 1999. ACM. ISBN 1-58113-096-1. doi: 10.1145/312624.312649. URL `http://doi.acm.org/10.1145/312624.312649`.

[23] Mark Jickling. Fannie Mae and Freddie Mac in Conservatorship. Congressional Research Service, Library of Congress, 2008.

[24] Kevin Knight and Jonathan Graehl. An overview of probabilistic tree transducers for natural language processing. In *Computational linguistics and intelligent text processing*, pages 1–24. Springer, 2005.

[25] Michael Laver, Kenneth Benoit, and John Garry. Extracting policy positions from political texts using words as data. *American Political Science Review*, 97 (02):311–331, 2003.

[26] Michael Levandowsky and David Winter. Distance between sets. *Nature*, 234 (5323):34–35, 1971.

[27] William Li, Pablo Azar, David Larochelle, Phil Hill, James Cox, Robert C Berwick, and Andrew W Lo. Using Algorithmic Attribution Techniques to Determine Authorship in Unsigned Judicial Opinions. *Stan. Tech. L. Rev.*, 16: 503–503, 2013.

[28] William P. Li, David Larochelle, and Andrew W. Lo. Estimating Policy Trajectories during the Financial Crisis. In *NLP Unshared Task in PoliInformatics*, 2014.

[29] William P. Li, David Azar, Pablo Larochelle, and Andrew W. Lo. Law is Code: Software Engineering the United States Code. In *Journal of Business and Technology Law*, 2015.

[30] Yu-Ru Lin, Drew Margolin, and David Lazer. Uncovering social semantics from textual traces: A theory-driven approach and evidence from public statements of US Members of Congress. *Journal of the Association for Information Science and Technology*, 2015.

[31] Robert V. Lindsey, William P. Headden, III, and Michael J. Stipicevic. A Phrase-discovering Topic Model Using Hierarchical Pitman-Yor Processes. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, EMNLP-CoNLL '12, pages 214–222, Stroudsburg, PA, USA, 2012. Association for Computational Linguistics. URL `http://dl.acm.org/citation.cfm?id=2390948.2390975`.

[32] Alexander Madrigal. Data.gov Launches to Mixed Reviews. `http://www.wired.com/2009/05/datagov-launches-to-mixed-reviews/`. Accessed: 2014-05-31.

[33] Qiaozhu Mei, Xuehua Shen, and ChengXiang Zhai. Automatic Labeling of Multinomial Topic Models. In *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '07, pages 490–499, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-609-7. doi: 10.1145/1281192.1281246. URL `http://doi.acm.org/10.1145/1281192.1281246`.

[34] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*, pages 3111–3119, 2013.

[35] David Mimno, Hanna M Wallach, Edmund Talley, Miriam Leenders, and Andrew McCallum. Optimizing semantic coherence in topic models. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, pages 262–272. Association for Computational Linguistics, 2011.

[36] Martin Moore. Churnalism Exposed. `http://www.cjr.org/the_news_frontier/churnalism_exposed.php`, 2011.

[37] Frederick Mosteller and David L Wallace. Inference in an authorship problem: A comparative study of discrimination methods applied to the authorship of the disputed Federalist Papers. *Journal of the American Statistical Association*, 58 (302):275–309, 1963.

[38] Ani Nenkova and Kathleen McKeown. A survey of text summarization techniques. In *Mining Text Data*, pages 43–76. Springer, 2012.

[39] Anh Phuong Nguyen and Carl E Enomoto. The Troubled Asset Relief Program (TARP) and the financial crisis of 2007-2008. *Journal of Business & Economics Research (JBER)*, 7(12), 2011.

[40] Barack Obama. Memorandum for the Heads of Executive Departments and Agencies: Transparency and Open Government. `http://www.whitehouse.gov/the_press_office/TransparencyandOpenGovernment`. Accessed: 2014-05-30.

[41] Barack Obama. Executive Order: Open Data Policy — Managing Information as an Asset. `http://www.whitehouse.gov/the-press-office/2013/05/09/executive-order-making-open-and-machine-readable-new-default-government-`, 2013. Accessed: 2014-05-30.

[42] Peter R. Orszag. Memorandum for the Heads of Executive Departments and Agencies: Open Government Directive. `http://www.whitehouse.gov/open/documents/open-government-directive`, 2009. Accessed: 2014-05-30.

[43] Daniel Ramage, Christopher D. Manning, and Susan Dumais. Partially Labeled Topic Models for Interpretable Text Mining. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '11, pages 457–465, New York, NY, USA, 2011. ACM. ISBN 978-1-4503-0813-7. doi: 10.1145/2020408.2020481. URL `http://doi.acm.org/10.1145/2020408.2020481`.

[44] David Robinson, Harlan Yu, William P Zeller, and Edward W Felten. Government data and the invisible hand. *Yale JL & Tech.*, 11:159, 2008.

[45] Jeffrey S Rosenthal and Albert H Yoon. Judicial ghostwriting: authorship on the Supreme Court. *Cornell L. Rev.*, 96:1307, 2010.

[46] Scout. The Sunlight Foundation. `https://scout.sunlightfoundation.com/`, 2014. Accessed: 2014-06-05.

[47] Jeong Seop Sim and Kunsoo Park. The consensus string problem for a metric is NP-complete. *Journal of Discrete Algorithms*, 1(1):111–117, 2003.

[48] Yanchuan Sim, Brice Acree, Justin H Gross, and Noah A Smith. Measuring ideological proportions in political speeches. In *Proceedings of EMNLP*, 2013.

[49] David A Smith, Ryan Cordell, Elizabeth Maddock Dillon, Nick Stramp, and John Wilkerson. Detecting and modeling local text reuse. In *Digital Libraries (JCDL), 2014 IEEE/ACM Joint Conference on*, pages 183–192. IEEE, 2014.

[50] Sunlight. The Sunlight Foundation. `https://sunlightfoundation.com/`, 2014. Accessed: 2014-05-31.

[51] Ben Taskar, Vassil Chatalbashev, Daphne Koller, and Carlos Guestrin. Learning structured prediction models: A large margin approach. In *Proceedings of the 22nd International Conference on Machine Learning*, pages 896–903. ACM, 2005.

[52] Joshua Tauberer. GovTrack.us. `http://www.govtrack.us/`. Accessed: 2014-05-28.

[53] Kiri L Wagstaff. Machine Learning that Matters. In *Proceedings of the 29th International Conference on Machine Learning (ICML-12)*, pages 529–536, 2012.

[54] Hanna M Wallach. Topic modeling: beyond bag-of-words. In *Proceedings of the 23rd international conference on Machine learning*, pages 977–984. ACM, 2006.

[55] Xuerui Wang, Andrew McCallum, and Xing Wei. Topical n-grams: Phrase and topic discovery, with an application to information retrieval. In *Data Mining, 2007. ICDM 2007. Seventh IEEE International Conference on*, pages 697–702. IEEE, 2007.

[56] John Wilkerson, David Smith, and Nick Stramp. Tracing the Flow of Policy Ideas in Legislatures: A Text Reuse Approach. *New Directions in Analyzing Text as Data. London School of Economics*, 2013.

[57] Tae Yano, Noah A. Smith, and John D. Wilkerson. Textual Predictors of Bill Survival in Congressional Committees. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, NAACL HLT '12, pages 793–802, Stroudsburg, PA, USA, 2012. Association for Computational Linguistics. ISBN 978-1-937284-20-6. URL `http://dl.acm.org/citation.cfm?id=2382029.2382157`.

[58] Bei Yu, Stefan Kaufmann, and Daniel Diermeier. Classifying party affiliation from political speech. *Journal of Information Technology & Politics*, 5(1):33–48, 2008.

[59] Harlan Yu and Stephen Schultze. Using Software to Liberate U.S. Case Law. *XRDS*, 18(2):12–15, December 2011. ISSN 1528-4972. doi: 10.1145/2043236.2043244. URL `http://doi.acm.org/10.1145/2043236.2043244`.