Distill-ery: Iterative Topic Modeling to Improve the Content Analysis Process

Chris Musialek
Department of Computer Science
University of Maryland, College Park, MD 20742
musialek@cs.umd.edu

ABSTRACT

Dimensionality reduction techniques such as Latent Dirichlet Allocation (LDA) provide extremely useful ways of making better sense of large, unstructured text data corpora quickly and efficiently. However, beyond small circles - typically comprising computer science and statistics experts - topic modeling has failed to become a commonplace technique incorporated into mainstream data analysis processes.

In this paper, we present Distill-ery, a visual topic model editor and an iterative LDA workflow that not only simplifies the consumption of the output of LDA, but also provides an easy mechanism to incorporate human-in-the-loop intuition into the resultant topics used to improve the quality of the model. We then use this tool to conduct two case studies in order to demonstrate the improved efficiencies of the approach and tool: one within a government agency known for its rigor of data analysis processes, the other within the domain of political science research. We discuss our approach in the context of other visual and iterative methods for developing and using topic models.

Keywords

LDA, topic modeling, computational linguistics, information visualization

1. INTRODUCTION AND MOTIVATION

Given the acceleration of the quantity of unstructured textual data produced within and across organizations and the continued desire to make sense of these data, computational techniques such as LDA show great promise for augmenting traditional content analysis processes of corpus exploration and categorization, many of which require time-consuming, manually-intensive effort. However, beyond small circles, topic modeling has failed to become a commonplace technique incorporated into mainstream data analysis processes, particularly among social scientists who analyze text.

```
[Topic 3: 257] help controller pay available cause create deal guy inactive modification
old_system rely Lockheed buy eram explain function i.e. legacy_system look

[Topic 4: 860] FAA hire technician people retire staff result leave plan staffing train
start manager training time lose process ability replace retirement
```

Figure 1: Typical output of LDA. Words of varying degrees of semantic coherence are grouped together with no textual context.

We surmise that this lack of uptick is is due to several factors. One, there is a lack of easy-to-use LDA tools built with the social scientist in mind. Most tools today require some programming knowledge, which significantly reduces their potential audience. Two, interpretability of topics is difficult. Most topic modeling software today simply outputs groups of words representing topics, which provides limited context for interpretation, as shown in Figure 1. In addition, incoherent words and topics often appear in the results. To date, we are not aware of work exploring how these approaches fare with social scientists in real-world use cases.

We present Distill-ery, an interactive visual topic model editor and workflow platform built with the social scientist in mind, that makes running, viewing, and interpreting the results from LDA extremely easy. Our prototype interactively displays important document context alongside the topics' word groupings to better inform a user of the most representative documents belonging to a topic, and highlights topic words within the raw text to make it easier to find relevant text of interest.

Additionally, we create a novel, iterative LDA workflow that permits incorporating human feedback from the output of LDA, to better inform a subsequent running of the LDA algorithm, biased by positive and negative word choices from the user. By easily allowing a user to tag words as coherent or incoherent within a resultant topic, to add custom words of their choosing from the corpus's vocabulary, and to merge and discard topics, we can assist the LDA algorithm to produce better results than would be possible using LDA alone. Described in more detail below, user feedback produces a new biased set of topic-word distributions that provide an informed initialization in the subsequent iteration of LDA.

To evaluate the effectiveness of the tool for assisting social scientists, we conduct two multi-dimensional, in-depth long-term case studies (MILC) [17], one with a professor of po-

litical science who is exploring how presidents talk about war and the degree to which war is framed differently, and the other with a methodologist at the U.S. Government Accountability Office (GAO) - which conducts independent, non-partisan policy studies for U.S. Congress - where we compare their traditional content analysis processes with the iterative topic modeling approach devised. Over the course of several months, we work directly with these participants, observing their behavior, and conducting interviews to record comments and other insights about the tool's effectiveness.

In the GAO case study, we use the same original data set from a completed GAO investigation, enabling us to do a direct comparison of our topic modeling methodology with their traditional content analysis process. GAO's investigations are known for their rigor of content analysis processes, and their investigations commonly analyze qualitative text data derived from in-depth interviews, focus groups, and expert forums. Because GAO has much to gain from improving their methodologies, they have begun to explore ways to find efficiencies in the analytic process while still maintaining the rigor and quality of the analyses.

Overall, our findings show promise for the iterative topic modeling approach, especially given the diversity of our case studies and their distinct analysis goals. Our work also has implications for both public and private-sector firms that focus on text-based content analysis, using text from speeches, news articles, focus groups, and in-depth interviews.

2. BACKGROUND

2.1 Topic Modeling Basics

Topic modeling refers to a family of computational techniques that extract latent topics or themes from collections of text, bringing to the surface underlying structure that is not likely to be immediately apparent. Variations of topic models have been used in the social sciences for diverse purposes, ranging from analyzing trends in the scientific literature, to characterizing patterns of agenda-setting and agenda-framing in political communication, to identifying thematic categories in psychotherapy transcripts. While our work strictly focuses on Latent Dirichlet Allocation (LDA), other approaches to topic modeling such as latent semantic analysis (LSA) [5] have been proposed before.

2.2 Latent Dirichlet Allocation

Latent Dirichlet Allocation is a probabilistic generative model proposed by Blei et al. [3] for collections of discrete data, typically text, to discover latent topics or themes within a corpus in an unsupervised way based on the co-occurrence structure of words. The general idea is that documents are represented as random mixtures over latent topics, where each topic is characterized by a distribution over words. The user decides a priori the number of topics that the model will attempt to fit.

LDA can be described more formally with the following notation. The topics are $\varphi_{1:k}$ where each φ_k is a distribution over the vocabulary. ϑ_m represents the topic proportions for the mth document; thus $\vartheta_{m,k}$ will represent the topic proportion of topic k in document m. We additionally define z_m as the topic assignments for the mth document, and

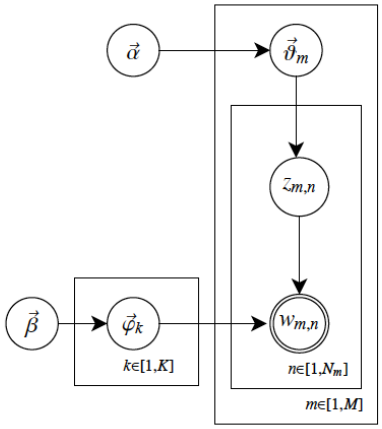

Figure 2: Plate Notation Diagram for LDA. The rectangles denote replication at each level. **Source:** ***Heinrich, Gregor. Parameter estimation for text analysis. 2005.***

M	number of documents to generate (const scalar).
K	number of topics / mixture components (const scalar).
V	number of terms t in vocabulary (const scalar).
$\vec{\alpha}$	hyperparameter on the mixing proportions (K-vector or scalar if symmetric).
$\vec{\beta}$	hyperparameter on the mixture components (V-vector or scalar if symmetric).
$\vec{\vartheta}_m$	parameter notation for $p(z\|d=m)$, the topic mixture proportion for document m. One proportion for each document, $\underline{\Theta} = \{\vec{\vartheta}_m\}_{m=1}^{M}$ ($M \times K$ matrix).
$\vec{\varphi}_k$	parameter notation for $p(t\|z=k)$, the mixture component of topic k. One component for each topic, $\underline{\Phi} = \{\vec{\varphi}_k\}_{k=1}^{K}$ ($K \times V$ matrix).
N_m	document length (document-specific), here modelled with a Poisson distribution [BNJ02] with constant parameter ξ.
$z_{m,n}$	mixture indicator that chooses the topic for the nth word in document m.
$w_{m,n}$	term indicator for the nth word in document m.

Figure 3: Quantities for LDA. **Source:** ***Heinrich, Gregor. Parameter estimation for text analysis. 2005.***

$z_{m,n}$ as the topic assignment for the nth word in document m. Last, we define w_m as the observed words for document m, and $w_{m,n}$ as the nth word in document m. Each of these words belong to the corpus vocabulary, or V.

This notation forms the three levels in the LDA representation: corpus-level parameters, document-level variables, and word-level variables. This is visually represented via the plate diagram in Figure 2, with quantities described in figure 3.

The observed and hidden variables are intertwined with one another, and together they define the *joint probability distribution* over both observed and hidden random variables. Using this distribution and Bayes' rule, we can define the *posterior distribution* of the hidden variables (the topics) given a document. Unfortunately, calculating the posterior distribution is generally intractable [Dickey, 1983] [6]. Therefore, an approximate inference algorithm such as Gibbs sampling [7] is often used, although other alternatives to Gibbs sampling exist, e.g. variational inference. Gibbs sampling is a special case of Markov-chain Monte Carlo (MCMC) simulation, which generates streams of data to estimate the needed distributions and give us a good approximation of the posterior distribution instead.

The overall LDA generative model can be described as follows:

1. **for** all topics $k \in [1, K]$ **do** //topic plate
   - sample mixture components $\varphi_k \sim Dir(\beta)$
2. **for** all documents $m \in [1, M]$ **do** //document plate
   - sample mixture proportion $\vartheta_m \sim Dir(\alpha)$
   - sample document length $N_m \sim Poiss(\xi)$
   - **for** all words $n \in [1, N_m]$ in document m **do** //word plate
     - choose a topic $z_{m,n} \sim Mult(\vartheta_m)$
     - choose a word $w_{m,n} \sim Mult(\varphi_{z_{m,n}})$

2.3 Topic Model Usage in Social Science

To inform our design, we spoke with data scientists at Reuters, an industry leading, international news agency, which recently utilized LDA to assist analysis of a special report [16] written on the business of practicing law before the U.S. Supreme Court. To produce their analysis, they programmed custom code leveraging a prominent, existing LDA tool called MALLET [13]. In our discussion with them, we came away with two key takeaways: one, the current state of LDA tools require programming skills, which is a major limitation in the journalism space, and two, LDA does not always produce coherent topics, which limited their use of the topic modeling approach.

As a second example, Quinn, et al. [15] describe a method for legislative speech based around topic modeling. Using LDA, they infer the relative amount of legislative attention paid to various topics within speeches in the U.S. Senate from the period of 1995 to 2004 (the 105th to the 108th Congress), at a daily level of aggregation. Additionally, they present several ways that social scientists can interpret and validate the results from the topic model, which has helped to guide our thinking in developing Distill-ery.

As a third example, Arnold, et al. [2] apply topic modeling inside the clinical medicine domain in order to automatically summarize clinical reports for primary care physicians. In it, they assess the coherence of topics and the extent to which they can represent the contents of clinical reports. Overall, they find that interpretable topics capturing specialized medical information can be discovered, which is encouraging.

Different approaches to improving content analysis have been proposed and explored before. Grimmer and Steward's [8] four principles of quantitative text analysis are excellent starting points for those interested in avoiding the pitfalls of automatic content analysis.

Grimmer and Steward's
Four principles of quantitative text analysis

1. All quantitative models of language are wrong - but some are useful.
2. Quantitative methods for text amplify resources and augment humans.
3. There is no globally best method for automated text analysis.
4. Validate, Validate, Validate.

Our attempt at improving efficiencies center around these ideas, in particular that the tools should augment the analyst's abilities rather than replace them.

2.4 Topic Model Visualization and LDA Improvement

Some work has been done to attempt to make LDA more accessible to a broader audience. Kim [11] has built a machine learning tool called Refinery that allows a user to upload documents, run topic modeling on the text, and visualize the results. One particularly nice feature is the ability to re-run LDA on a subset of the documents based on some initial filtering (e.g. all highly probable documents associated with an original latent topic). However, they do not provide a way to improve the initial model based on human-in-the-loop feedback, as we propose.

Additional work has been done to optimize semantic coherence of LDA through an interactive topic modeling approach. Boyd-Graber et al. [9] developed a framework to allow users to interactively refine topics discovered by LDA by adding constraints that enforce that sets of words must appear together in the same topic. In their motivating example, they attempt to converge two topics that both deal with Russia, one about the Soviet Union, and the other about post-Soviet years. To do so, they manually created a constraint with all of the clearly Russian or Soviet words, which led to the two topics being combined after a subsequent set of iterations, with one of the original topics now more about elections in countries other than Russia.

A drawback to their approach is the issue of transitive convergence. A constraint between Russia and Soviet Union could also link a third or fourth, yet semantically distinct, topic containing the same word Russia, but used in a different context. Instead of maintaining distinct topics, which one would expect, their constraint model force these topics together, which is unlikely to increase semantic coherence.

Previous work has also been done to study the effects of how non-expert users perceive topic models, and the potential benefits of allowing them to make refinements to topics. Lee et al. [4] ran a formal experiment with college students to assess the quality of the topic as well as whether adding or removing words from the topic made them more coherent. To a great extent, what we found is consistent with their more formal study, and what we each did is complementary. However, while they focus on a constructed (but realistic) scenario with college students, by contrast, we conducted case studies with real social science experts looking at real-world data and use cases.

3. INTERFACE DESIGN AND FEATURES

Below we describe several of the features we designed into Distill-ery to assist with our iterative topic modeling approach. They are organized by the two major screens that form Distill-ery: the project screen, and the individual model screen. Below we describe each in more detail.

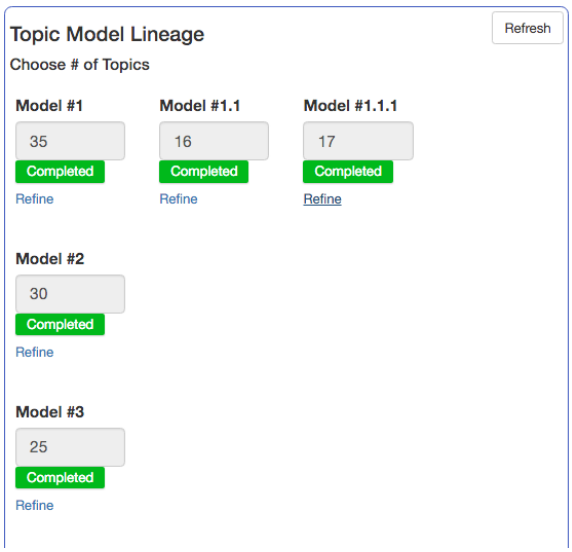

Figure 4: The project screen's topic model lineage box. It allows the social scientist to quickly view the progression of model refinement using Distill-ery.

Figure 5: Each model ID inside the Topic Model Lineage box provides a status indicator of the current state of running LDA. A "Refine" link pointing to the model results (the Individual Model Screen) appears upon completion.

3.1 Design Goals

We had several major design goals in mind. One, because social scientists are unlikely to have programming skills, it was important that Distill-ery automated every aspect of LDA - from uploading data to running LDA to adding new models - in a simple and easy to follow workflow. As such, all interaction is GUI based, and all LDA details are handled behind the scenes using a custom-designed, Web API.

Data security and data privacy was another key design goal. Social scientists' data are often sensitive and cannot be shared with outside parties. As such, we designed Distill-ery to be easily set up and run locally on their laptop or work computer so that no data is ever passed to a third party; all computation occurs locally. It is also straightforward to set up a private install on Amazon Web Services (AWS) for each specific user. In our two case studies, one (GAO) used AWS, and the other (political scientist) ran the software locally.

3.2 Project Screen

The project page encapsulates the entire workflow at a glance, and is the central interaction point to perform all major ac-

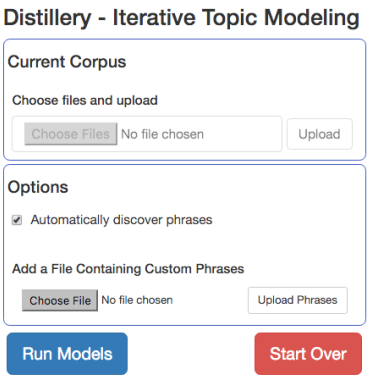

Figure 6: Corpus text is easily uploadable, and Distill-ery provides pre-processing options to automatically discover phrases and/or upload a custom set of phrases. The social scientists in our study both specifically requested this. It is also extremely easy to run your models, or startover.

tivities on a corpus, including uploading the corpus to the application, pre-processing the data according to user defined settings, and creating and running LDA models themselves.

On the left-hand side of the display, the user can easily select raw text files to be uploaded to the application. In the options box shown in figure 6, we provide two customizable pre-processing options to the user: automatically discover phrases, and upload a custom set of phrases. The automatic phrase discovery feature is an implementation of Justeson and Katz, 1995 [10], who developed a novel regular expression based on the words' part-of-speech to combine words into likely technical phrases.

To pre-process the raw text, Distill-ery leverages CoreNLP's [12] tokenization and lemmatization features to create the vocabulary necessary for running LDA. Each token in Distill-ery is the predicted lemma from the output of CoreNLP's pipeline. Prior to lemmatization, phrases are discovered and converted to tokens based on the above methods.

On the right-hand side of the display we present all models together at a glance, using a simple lineage metaphor to assist the user understand how models relate to one another within the overall workflow. At the bottom of the display, the user can easily click a button to create a new "root" model at will. A root model is any model which has no parent, and thus, no informed initialization. See section 3.5 for more details. All child models are created using informed initialization based on the output of a user's interaction with its parent model. Below, we will describe this interaction in more detail.

The overall idea of our LDA workflow is to allow users to quickly gauge how well a specific choice K for number of topics works for the corpus, as well as to iteratively incorporate human-in-the-loop feedback from the output of each LDA iteration which can be used to assist the running of LDA in a subsequent iteration by use of an informed topic-word

initialization.

Because running LDA can take time depending on the size of the corpus, we provide a status indicator under each model with the number of iterations completed to give the user a sense of how far along each model is. Once LDA is completed running, the tool will display a link to the individual model screen which shows the results of LDA and premits user feedback.

3.3 Individual Model Screen

Figure 9 presents the results of a single LDA run. It allows the user to interact with these results and provide human-in-the-loop feedback to assist a subsequent running of LDA.

On the left-hand side of the display - using the generated topic-word distributions created from the model - we display the top-N words associated with each topic in an easy-to-read scrollable pane. These top-N groupings of words represent the most probable words given the topic ($Pr(w|t) = \varphi_t$). We have additionally provided a feature that allows the user to select N (between 10 and 50 words), in order to provide some flexibility of the display of words.

To assist the user associate a topic with a semantic theme, we give them the ability to add a custom label to each latent topic. This enables them to capture the essence of the topic's meaning in a single phrase for the user, and allows them to return to each of the topic words at a later time and quickly recall the overall semantic theme of the topic without expending further effort. We also carry over these labels from iteration to iteration using a simple heuristic algorithm we developed which attempts to match the words associated with the existing label from the most recent topic editing round.

When a user uses the same custom label in multiple topics, this indicates to Distill-ery that the topics are identical and should be merged in the next iteration.

At the top of the left-hand side of the display, we provide two additional buttons: an "Export" button, and a "Recalc" button. The *Recalc* button is used to tell Distill-ery that the current set of confirmed and rejected words are satisfactory, and that Distill-ery should convert these word selections into a new set of topic-word distributions ($\varphi_{1:k}$) and create a new model using this custom initialization. We describe the conversion of selected words to topic-word distributions in Section 3.5.

The *Export* button allows the user to download the raw document-topic distributions in .csv format, with headings based on the custom label chosen for the topic.

On the right-hand side of the display - using the generated document-topic distributions created from the model - we display the top-M documents associated with each topic as a bulleted list within an independently scrollable pane. These top-M documents are simply the full raw text of each document, ranked by their individual probability weight within a topic, and filtered by the top-M examples as a threshold. For long documents we show just the beginning of the document, and the user can expand/reduce to inspect the full document if desired. When the user hovers over a new topic grouping on the left-hand side of the display, a new set of top-M documents appear, easily allowing the user to browse the individual documents that are most likely to be associated with the latent topic and the words generated alongside it.

[1]Our user gave us feedback on this feature, but did not actually use it in the specific iterations described in Section 4.

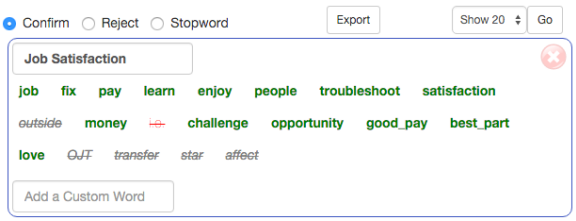

Figure 7: Distill-ery allows the user to confirm or reject words in a topic, add stopwords, label topics, and assign custom words from the corpus vocabulary.

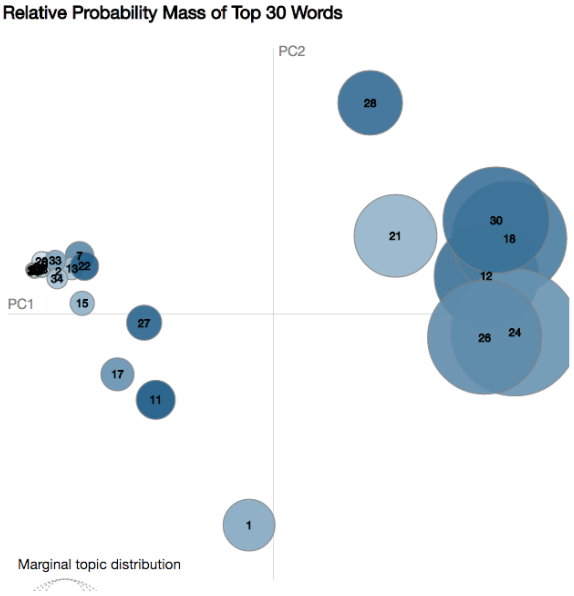

Figure 8: Intertopic Distance Map [1]

Additionally on the right-hand side of the display, we provide a tab to an intertopic distance map view which presents a grid showing all topics as circles, as shown in figure 8. This feature was inspired by LDAvis [18]. The placement of the circle on the coordinate system is calculated using Principal Coordinate Analysis (PCoA), which allows us to visualize the relative distance between each multinomial topic distribution. Relative probability mass of the top N words for each topic is also shown using color shades. When a user hovers over a single word in a topic, we show that word's relative word-topic proportion on this graph. For example, in Figure 8, the user can see that topics 30, 18, 12, 26, and 24 are similar, which might lead to inspecting them for

comparison and possibly giving them the same label, which automatically merges them.

To provide even more context in understanding the topic, when the user hovers over an individual word inside a topic grouping, we highlight the word within the text for them, and additionally filter documents containing at least one appearance of the selected word, as shown in Figure 9. This interaction permits the user to easily reason *why* each top-N word was generated within a topic, and also better understand the word's most likely word sense, by reading the phrases and sentences surrounding the word. Displaying the raw text alongside a latent topic is important for making better sense of the limited information afforded the user by the top-N words alone.

Distill-ery also allows the user to create their own customized popup box that displays customized HTML when hovering over any chosen word. The idea behind this popup box is to provide useful context about the word, such as the word's sentiment, word's translation, and word's sense. The popup box supports any valid HTML at this time.

3.4 Informing the Model

A principal goal of our workflow is to use human intuition to assist the model in producing more coherent topics, and to align them with the best set of documents in order to automatically tag those documents. Several features facilitate this process.

The user can easily tag specific words within each individual topic as coherent, not-coherent, or a stopword. Using the radio buttons at the top left-hand side of the screen to toggle the type of tagging, clicking on an individual word will highlight it as either green and bold for confirm, gray and strikethrough for reject within this topic, and red and strikethrough to set this word as a stopword across all topics.

If the user would prefer to add additional words not present in the current list of words to strengthen the topic, Distill-ery provides the ability to add a custom word from the vocabulary. The custom word text box suggests vocabulary words as the user types, as shown in Figure 10.

Confirmed words

Confirmed words display as **bolded green**. They mean the word is coherent within the topic's overall theme. These words become part of the topic's new seed probability mass.

Rejected words

Rejected words display as ~~*strikethrough italicized gray*~~. They mean the word is incoherent within the topic's overall theme. These words, along with all other non-selected words in the vocabulary, become part of the topic's non-seeded probability mass, explained in further detail in section 3.5.

Stopwords

Stopwords display ~~strikethrough red~~. They mean the word should be omitted from all topics' probability mass, and given a probability of zero across all topics.

Additionally, we provide a clickable red "X" button at the top right-hand side of each topic (shown in figure 7) to allow the user to completely reject an entire topic as incoherent. Sometimes, LDA produces topics whose generated words and most probably associated documents bear little relationship amongst each other. By providing a way to ignore an incoherent topic, we provide another means for the model to improve its results after another iteration of user feedback.

3.5 Transforming Tags to New Probabilities

Once a user is done labeling, tagging, and potentially deleting entire topics, we need to convert these human interactions into new, seeded probabilities that the model can use to give some advantage to these choices when the model is rerun. Below, we describe how we do this in detail.

The Segan [14] implementation of LDA provides the capability of initializing the model with an existing topic-word distribution. Rather than simply initialize each latent topic as a random distribution of words over the entire vocabulary, we instead seed these initial distributions based on the user's feedback. Each topic's initial distribution is created as follows:

Define C as the number of total seed, or "coherent" words tagged across all topics, S as the number of total stopwords tagged, and $N = |V| - C$, or the size of the vocabulary minus seed words. Additionally, let us define $0 < K < 1$ as the amount of probability mass being given to the seed words. (In our experiments we used K=0.80)

1. For each seed topic t:
   (a) For each seed word i in t, let
   $$base_i = K/C$$
   (b) For each non-seed word j in t, let
   $$base_j = (1 - K)/N$$
   (c) For every word w, create a "jittered" value, where ϵ is some small value, typically between $0 - 5\%$ of $base_w$:
   $$Pr_w = base_w \pm \epsilon$$
   (d) For each word in t, let
   $$Pr_{w|t} = \sum_{w' in words} Pr_{w'}$$
2. For each stopword s, let
   $$Pr_s = 0$$

Figures 13 and 14 display coherent and incoherent words within a topic before and after running an additional iteration based on this method. After running an additional iteration with the custom initialization, the coherence of the topic improved.

It is also important to note that we only custom create a topic's initial distribution if the topic was considered coherent. Currently we define those as topics having at least one coherent (confirmed) word in it. For all remaining topics, we simply use a random initialization. When the new

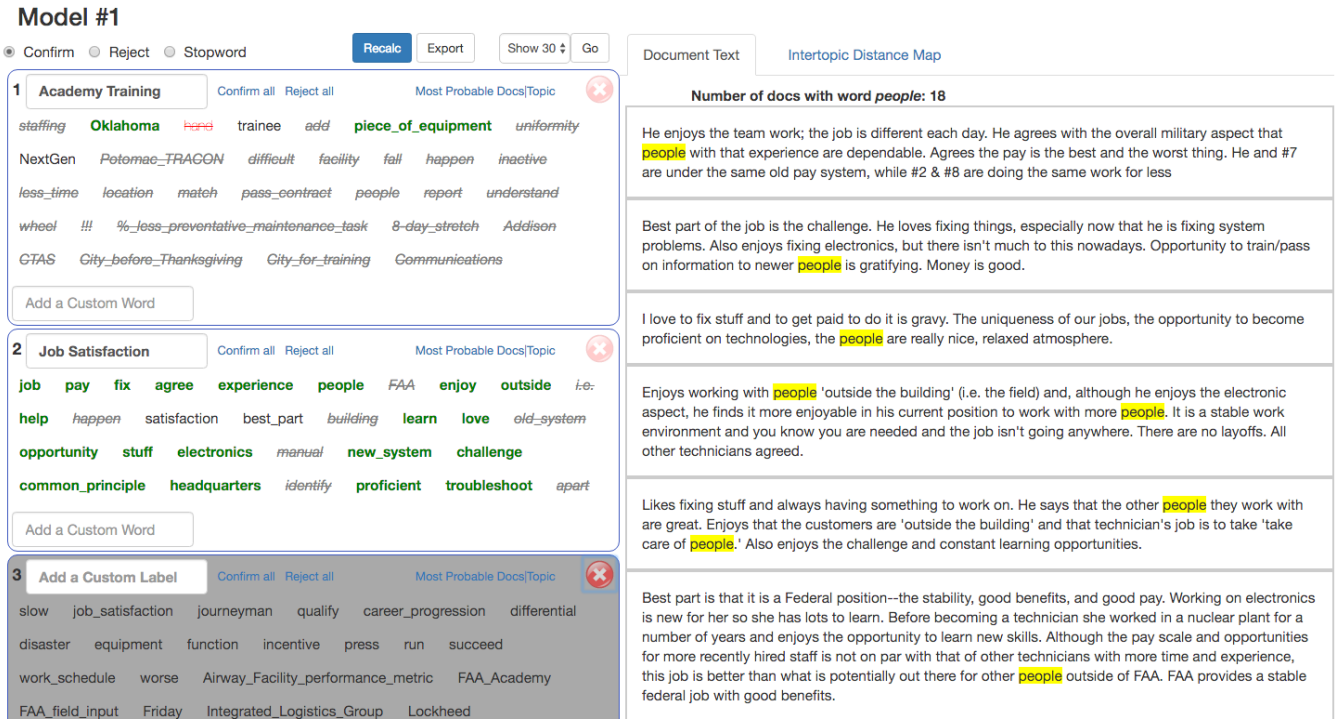

Figure 9: When a user hovers over a word, the raw text is filtered and all instances of the word are highlighted in yellow.

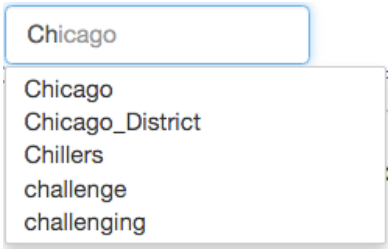

Figure 10: Adding a custom word will suggest words from the vocabulary as you type.

model shows up in the Project Screen, Distill-ery will suggest a new K using the number of coherent topics + 5 as a baseline. The user is welcome to change this suggested K before actually running the new iteration.

3.6 A Proposed New Process

Figure 13 provides an overall flow diagram that describes how we incorporate Distill-ery into an iterative topic modeling process. Once LDA has been initially run, we load the results into Distill-ery, and allow the user to select their choices of seed words, non-seed words, and new stopwords, shown in yellow diamond #3. In addition, labeling topics with the same label will merge them into a single topic in the next iteration of the model. Entire topics can also be deleted easily.

Once the user is done making changes to the individual topics, this human feedback data is converted into new, artificial topic-word distributions as described above, and then used as an informed initialization when re-running LDA. This process loop (yellow diamonds #2-4) can continue as many times as the analyst would like until s/he is satisfied with the final results.

The final stage in the process is, for each coherent topic, to auto-tag documents with the highest probability given the latent topic, using a percentage threshold as a cutoff. Documents that remain un-tagged will then be forwarded on for manual review as they normally would in any existing content analysis process. Figure 14 visually expresses this final component.

4. CASE STUDY #1: ANALYSIS OF FOCUS GROUPS

Case study #1 was conducted with a content analysis expert inside the U.S. Government Accountability Office (GAO), using the same original data set from a completed GAO investigation conducted in 2010. This allowed us to do a direct comparison of our topic modeling methodology with an example of their traditional content analysis process.

GAO had several goals in mind in exploring a topic modeling methodology with us. GAO's existing investigation methodology for content analysis is very time consuming, requiring between 250-320 person-hours for each report, which has the effect of limiting the number of case studies that GAO can perform given current staffing levels. Much of the effort spent in the existing process, which is explained in detail

in Section 4.1, is due to the manual process of reading all transcription text taken from in-depth interviews in order to discover common themes among them, building a taxonomy of the responses, and categorizing the original documents accordingly. If GAO could replace parts of the manual process of theme discovery and document tagging with topic modeling while still maintaining the current rigor and quality of the analyses, they could conduct more investigations without additional staff.

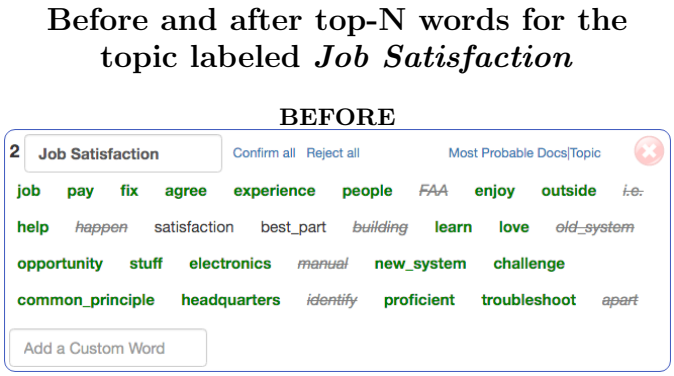

Figure 11: Words selected by a GAO expert of a coherent topic prior to re-running LDA using our process.

Figure 12: After iterating, more of the top-N words belonged to the topic, indicating improvement.

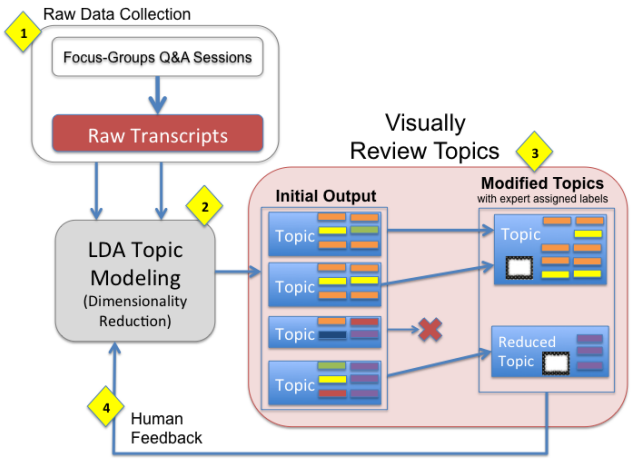

Figure 13: Workflow for content analysis using iterative topic modeling.

4.1 A Rigorous, Real World Process

Because GAO conducts independent, non-partisan policy studies for U.S. Congress of agencies inside the executive branch of government, their studies often rely on in-depth interviews, focus groups, and expert forums to collect raw data and inform their analysis. Below we briefly describe an existing study performed in 2010 [GAO-11-91] [19] by GAO involving the Federal Aviation Administration (FAA).

For this particular study, the Chairman of the Subcommittee on Aviation, Committee on Transportation and Infrastructure in the U.S. House of Representatives asked GAO to investigate the FAA's air traffic control (ATC) equipment outages and failures, which had potentially serious air safety implications. Since 2006, ATC equipment failures were blamed for causing hundreds of flight delays, and these delays raised questions about FAA's maintenance capabilities [19]. About 5,100 technicians maintain FAA's current (legacy) facilities and equipment and will be responsible for the Next Generation (NextGen) technologies planned for the next 15 years. Thus, much of the focus revolved around technician training practices, and how the FAA's key practices compare to those of leading organizations.

More specifically, GAO was requested to review:

1. How FAA incorporates key practices of leading organizations in its workforce planning for technicians.
2. How FAA's technician training compares with key practices of leading organizations.
3. How the costs of technician training, including travel costs, have changed in recent years.

In order to assess such questions, one of the methodolo-

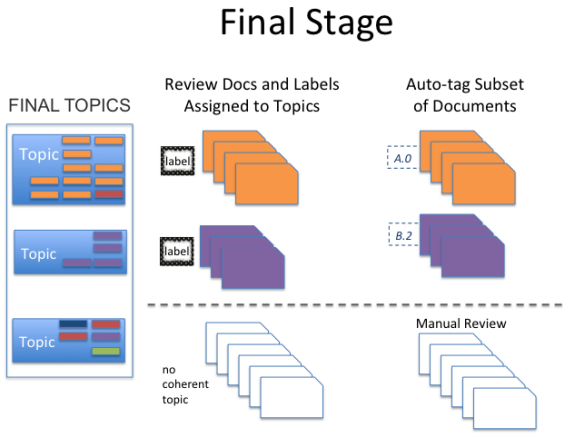

Figure 14: The last stage is to auto-tag documents. Remaining documents are tagged manually.

gies chosen by GAO was to conduct a series of focus groups with FAA employees who are most familiar with the current practices of the organization, and they conducted focus group with these individuals, asking them a series of questions, and recording their answers. For the FAA report, GAO conducted focus groups with FAA technicians and FAA Training Academy instructors in twelve different ATC centers and training centers around the country. Each focus group interview lasted approximately 90 minutes, and included three to ten FAA employees in various roles and with varying degrees of experience. Present as well were the GAO staff, whose job it was to administer the questions and scribe the responses for later analysis.

GAO analysts created a standard set of focus group questions to be used consistently across all ATC centers, and organized each focus group session around similar categories of questions. For each question, the moderator would go around the table once and get his/her answer before then opening up the discussion to everyone.

The study collected a total of 1082 individual responses across twelve ATC centers visited. Each response varied from a minimum of two words to a maximum of 366 words, with a mean of roughly 41 words, for a total of 44,575 words. The reader will note that relatively speaking, this is a fairly small amount of text, but this is common in real world settings.

Once the raw data is collected and transcribed, the first key task of GAO's analysts is to carefully read all responses individually and write down top keywords and phrases that they feel best summarize and capture the interviewees' answers for each question category. For example, for the question category entitled Training Issues, notes included "insufficient training on-site", and "lack of training on legacy systems".

These notes are compiled across the multiple analysts manually reviewing the text responses. Next, the analysts work to build a taxonomy of responses, categorizing the keywords and phrases into well-defined categories and sub-categories. For example, for the above notes, the former was categorized under "On-the-Job-Training", and the latter under "General lack of overall training on systems". These labels are then converted into codes based on their hierarchy in the taxonomy, and then two analysts read through each response again, tagging them independently with the appropriate labels. If two analysts disagree about the labels, a third analyst weighs in to resolve the disagreement.

In all, this process took about 320 person-hours to complete for this report, a material amount of time and effort.

4.2 Data Preparation

GAO focus group records were initially provided to us in an excel spreadsheet, one worksheet per focus group, and within each worksheet, one row per response. Each row included the raw transcript text plus multiple additional columns of metadata about each respondent (such as years of experience, role, etc) as well as the codes given to each response by GAO's analysts.

The actual language in this dataset involved notes from the scribes, not exact transcriptions of the employees' speech. In several cases, the scribes attributed the exact textual response to several people in the group. This occurred when everyone in the interview had stated similar things. Because of this, duplicate text responses existed, which we decided to eliminate in order to avoid over-biasing the model in favor of these duplicate scribed responses. After pre-processing to remove duplicates, we were left with 994 responses, from an initial 1082 responses.

4.3 Case Study Process

Over the course of two months, we worked directly with our GAO analyst to assess our approach. Below, we describe a single experiment with our participant.

We stood up a Distill-ery instance on Amazon EC2 for our participant, and after some initial guidance about the basic features of the tool, we asked them to upload their raw documents and begin iterating. Our GAO analyst began with 25 topics, but discovered quickly this was not enough, then added a new "root" model with 30 topics, and ran again with ease. Finally, based on the outputs of these first iterations of LDA, the analyst settled on 35 topics.

Having settled on 35 topics, our analyst ran through several iterations of LDA without our intervention, confirming coherent words, and rejecting both incoherent words and stopwords. Overall, he kept only 11 of the overall topics, marking the other remaining as incoherent. He individually labeled the 11 topics, and marked some of the words in each. Figure 11 displays the confirmed words for the topic manually labeled *job satisfaction*.

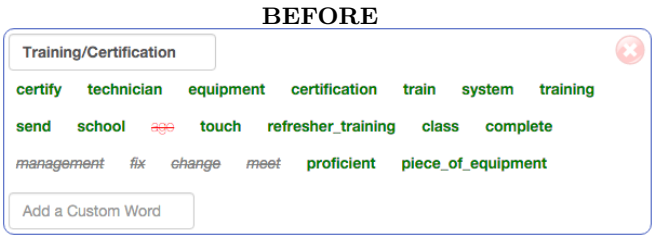

Figure 15: Our GAO analyst's initial label for the topic, which concentrated on FAA management issues.

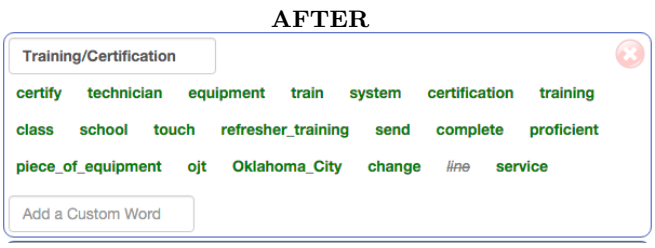

Figure 16: After iterating, a more nuanced interpretation of the topic emerged, leading the GAO analyst to re-label it as Mgmt/Union Relations.

After a recalc, new results were automatically loaded for the user. He then made another pass at tagging the topics. The

results of the next iteration for one "job satisfaction" topic are shown in figure 12.

We also asked him to provide a qualitative judgment as to whether the iteration improved the quality and precision of the topics. He judged that this single iteration did improve the quality of the topics quite a bit. Words such as "building", "old_system", "apart", and "i.e" are no longer there, and additional words such as "autonomy", "solve", and "love" appear, which semantically align.

Figures 15 and 16 also demonstrate improvement of the model using our iterative approach. Initially, the words "ago", "management", "fix", "change", and "meet" were considered irrelevant. After iterating once using distill-ery, additional words such as "ojt" (for on-the-job training) appeared. Again, the word "change" re-appeared, but in the subsequent iteration, the analyst changed his mind and considered it to be coherent.

In an earlier experiment where our user also discovered the theme "job satisfaction", the word "outside" was initially considered incoherent by the analyst, but was reintroduced into the topic after iterating. This represents a word - despite our strong bias down-weighing its importance - in which the evidence for this word in the data with the others in this topic was so great that it overruled our bias. Subsequently, the analyst decided the word indeed was sensible for this topic based on context. In the experiment above, the word "outside" was considered coherent from the outset.

4.4 User Feedback and Evaluation

4.4.1 Evaluation

Below, we consider a key question in comparing our approach to GAO's existing method: how close did LDA topic modeling get to replicating GAO's traditional content analysis original themes and categories?

Theme Detection. We asked our GAO analyst to take the 11 topic labels representing the coherent topics/themes from the final output of LDA, and map them to their respective categories in the original GAO investigation's taxonomy, shown below.

Based on this mapping, we calculated the overlap of the GAO categories discovered through LDA versus the total number of categories found from the original GAO investigation. Distill-ery found 116/124 of the original GAO categories, representing 93.5% overlap of the two. All topics/themes were discovered after the first iteration (running normal LDA), and remained throughout all subsequent iterations.

Coherence of Themes Across Iterations. We also asked our analyst to qualitatively measure the coherence of each discovered theme across iterations in order to determine whether our human-in-the-loop workflow was working. For each model in the lineage, he looked at the most probable documents within each coherent topic/theme he found, and ranked each document list on a scale of 1-10, where 1 means: does not at all reflect the theme, and 10 means: completely reflects theme.

Mapping of Topic Model Labels to Original GAO Categories

	GAO Topic Labels	Original Taxonomy Codes
1.	Academy Training	D7, D10, F1, F4, F5, F6, F8
2.	Job Satisfaction	A (A1 thru A9)
3.	Staffing Challenges	B2, B3, B7, B8, C1, C2, C3, E1, E2, E3, E6, E8
4.	Equipment Issue	B10, C6, D3, D4, E5, G (all)
5.	Shifts and Scheduling	B4, B5, B11, B12
6.	Tech Ops Pay/Funding	B6, C4, E4, E7
7.	Vendor Training	D8
8.	Refresher/OJT Training	D2, D6, F2, F3, F9
9.	Contractor Issues	C7
10.	NextGen Issues	C8
11.	Technician Issues	C5

*Taxonomy codes defined in Appendix A.

Table 1: Mapping of discovered topic labels to GAO's original taxonomy codes

Overall, there was an increase in the coherence of every discovered theme based on the documents most highly associated with the topics, as shown in Table 2.

In addition, there also was an increase in the coherence of almost every discovered theme based on the top words representing the topic, as shown in Table 3.

4.4.2 User Feedback

The tool's automation is key. Our user particularly liked how easy it was to get up and running with Distill-ery, especially how easy it was to create new models. He noted that it was important for him to be able to run through an experiment by himself, which he did without issue.

Providing textual context was extremely important. Our GAO analyst stated that he regularly used both the list of documents pertaining to each topic, as well as the word highlighting feature often in helping him determine the coherence of a word, as well as to discover other related words to add to the topic.

User expressed uncertainty as to what denotes completion. Our participant was not entirely certain how to best determine when he should consider himself done. Over time, he grew to like the intertopic distance map as a potential means to determine completion, as over successive iterations, it appeared that the bubbles in the graph would successively separate more widely.

Coherence of Themes by Quality of Document List
(1 is low, 10 is high)

	GAO Topic Labels	Iteration 1	Iter. 1.1
1.	Academy Training	3	9
		5	
		9	
2.	Job Satisfaction	8	10
		4	
3.	Staffing Challenges	9	9
		6	
4.	Equipment Issues	7	9
		5	
5.	Shifts and Scheduling	7	9
6.	Tech Ops Pay/Funding	7	8
7.	Vendor Training	5	9
8.	Refresher/OJT training	8	8
		7	
		7	
9.	Contractor Issues	6	9
10.	NextGen issues	7	9
11.	Technician Issues	7	8

Table 2: Iteration 1.1 shows improvement from Iteration 1 across all coherent topics/themes found. Multiple values per label indicates multiple LDA topics were given the same label, and then merged in a subsequent iteration.

Coherence of Themes by Coherence of Words
(1 is low, 10 is high)

	GAO Topic Labels	Iteration 1	Iter. 1.1
1.	Academy Training	3	10
		6	
		7	
2.	Job Satisfaction	8	10
		4	
3.	Staffing Challenges	6	10
		6	
4.	Equipment Issues	7	10
		5	
5.	Shifts and Scheduling	6	9
6.	Tech Ops Pay/Funding	6	8
7.	Vendor Training	6	9
8.	Refresher/OJT training	8	9
		8	
		6	
9.	Contractor Issues	7	10
10.	NextGen issues	7	9
11.	Technician Issues	7	7

Table 3: Iteration 1.1 shows improvement from Iteration 1 across all coherent topics/themes found. Multiple values per label indicates multiple LDA topics were given the same label, and then merged in a subsequent iteration.

5. CASE STUDY #2: ANALYSIS OF POLITICAL SPEECHES

We worked with a professor of Political Science in our second case study, who is exploring how presidents talk differently about the costs of war. Her goals, which we describe below, are much different than GAO's.

First and foremost, our user would like to identify several topics/dimensions on which leaders of different types might vary (e.g. war costs, discussion of victory, etc). Her expectation is that some of these will be created directly based on theoretical expectations while others will be suggested by the data.

Second, our user would like to determine if culpable leaders differ from non-culpable leaders in how they employ words associated with these topics. For example, do non-culpable leaders talk about the costs of war more than culpable ones? Do culpable leaders talk about victory more?

Culpable leaders are defined as individuals who were in charge at the start of a war, or those who followed a culpable leader and who share a clear link with the culpable leader, such as political connection, or belonging to the immediate ruling group.

Alongside these goals, she already had a specific hypothesis in mind: if a president frequently talks about victory in speeches, citizens are likely to be more tolerant of the costs of war. A president hoping to frame war in terms of victory, might decide to use "sacrifice" and "service" over "casualties" or "wounded".

Thus, the desire for LDA for their use case is to identify and quantify specific evidence in presidential speeches indicating the degree to which presidents framed the discussion of the costs of war differently depending on whether they are on the winning or losing side of the actual outcomes of the war.

Unlike GAO, this user is not interested in tagging specific documents. Instead, she hopes topics will emerge from the output of LDA that nicely separate the presidents into two distinct categories. In addition, because she already has specific topics in mind that she posits might exist, she's interested in giving LDA a starting point via a set of topic of words that *should be* present, using the algorithm to validate or invalidate those biases.

5.1 Current Content Analysis Methods in Political Science

Political scientists are already using a variety of techniques for content analysis. One of the more common methods today is called the dictionary method [8], whereby key word counts of two dichotomous lists of words are calculated for each document. Common uses of this are to categorize a corpus of documents into two groups, such as positive and negative campaign advertisements.

Supervised learning methods are also used today in political science, but they appear to be limited in scope due to the high cost of hand classifying a training set on which the model relies on.

Unsupervised techniques such as topic modeling are also being utilized as a content analysis method in political science. Political scientists have extended topic models so that the parameters correspond to politically relevant quantities of interest [8], such as the dynamic multitopic model, and the expressed agenda model.

LDA is also being used today by political scientists to do automated content analysis. However, its use is limiting in the sense that uses of it today do not make room for human-in-the-loop interaction. To date, political scientists have not had LDA tools which provide an input mechanism from the user into the unsupervised model. In our case study, we assess a more iterative approach to LDA for the analysis of political content.

5.2 Case Study Process

In this case study, we worked directly with our user and her student to assist them in answering their specific research questions. Over the course of several months, we interacted with them regularly, answering questions about the software, asking them to write up thoughts on the software, and also notes helping to better understand their use case. In addition, we provided advice for how to best utilize iterative topic modeling to further her specific research goals.

5.2.1 Data Sources

The corpus of data we used was screen-scraped from the American Presidency Project [1], and provided to us as several directories of text files, one directory for each president, and one file for each speech. Each file contained metadata about the speech as well as the speech itself, so we were required to pre-process these text files to extract the specific transcribed text of the speeches. We also were given a directory of filenames in an excel spreadsheet, which had other important information to the researchers, such as date of the speech, source, and the speech title. For this case study, we only used speeches from two U.S. presidents: George W. Bush, and Barack Obama.

The total size of the corpus used in our case study includes:

- 5,756 George W. Bush speeches from 2001-2009.
- 4,028 Barack Obama speeches from 2009-2014.

Our user ran Distill-ery locally on her laptop, and had no issues uploading her documents into Distill-ery. She initially ran LDA against all documents in the corpus, which took many hours due to the quantity of documents and the length of each speech. For additional purposes of trying out the interface and more quickly learning about how well the iterative approach worked, she created a separate instance of Distill-ery, and loaded about 90 documents into it, which she felt was enough to assess the tool. From this, she ran an initial LDA iteration with 10, 20, and 30 topics, and settled on 20 topics as the best number.

For each topic, they tagged coherent and incoherent words, and added custom words that they felt associated with the topic.

In addition, they attempted to create a completely custom topic from scratch. We were not expecting this, and so we had not built in an intuitive way to create a completely new custom topic. To compensate for this lack of functionality, we instructed these users to choose any incoherent topic, reject all existing words, and then add individual custom words that they desired. While un-intuitive, these actions functionally produced the same informed initialization, which was the goal. In a future iteration of Distill-ery, we plan to provide a "create custom topic" capability.

Quote from our user:
"This [topic] was AWESOME. Really impressed with how quickly this topic came together. This was either the very first time we saw the topic or one 'revision' in"

5.3 User Feedback

Below we describe some of the major themes we discovered during our case study.

The tool must automate all aspects of LDA. Originally, our software required editing a configuration file, as well as running a python script that produced the output required to power the Individual Model Screen. One of the major pieces of feedback we received in this case study was that automation was absolutely critical. In order for a typical social scientist to use it, these users felt that the entire process of running LDA, including uploading files, running the models, and seeing the results, absolutely needed to be GUI based. This feedback led us to build a full client-server architecture with a java backend containing a powerful API that not only ran the models for the user, but also kept track of all state information from the generated human-in-the-loop feedback.

Provide instant LDA status feedback to the user. Because LDA can take a while to process depending on the size of the corpus, this case study's users asked for a progress indicator in the topic model lineage box to help them gauge the current state of execution, which we added during the study.

Informed initialization of LDA shows great promise. This case study's users consistently were happy with the improved results of LDA after a subsequent iteration using an informed initialization built from the confirmed set of words in the previous iteration.

Use custom initializations to build hand-crafted topics. This case study's users consistently asked us how to create their own custom topics using words they hoped would show up together. Because their goal was to find evidence of how leaders speak differently across several topics/dimensions that they already had in mind, they wanted to pre-create

topics that LDA would use to attempt to find evidence of such a topic. Then, ideally, they hoped to see how the different leader speeches ranked within each topic. This is an interesting use case for LDA we had not anticipated.

Provide transparency into the entire process. Another critical element for social scientists is the importance of replicability and documentation at each step of the process. For many social scientists, statistical techniques such as LDA are not well understood, and therefore receive additional scrutiny from peers. One key piece of feedback was the desire to create a log file of everything done by the user, in order to document "here's what I did to get to these results".

In addition, these users desired to better understand how other parts of the tool worked, such as: how does the automatic phrase detection work? Furthermore, they wanted to see a list of the specific vocabulary that Distill-ery created from the text, which we added as a feature of the export.

Should we start narrowly, or broadly?. Another question that continually surfaced had to do with the level of breadth of data to start with. The users in this case study wanted to know whether it was a good idea to first pre-filter speeches by a manually determined set of terms about the costs of war instead of starting with the entire set of speeches. Our advice to them was to start broadly, using LDA to first attempt to discover themes about war, and then use the most probable documents out of the first process as filtered subset for running LDA again. This was not directly apparent to these users, which probably is due to a lack of understanding about how LDA works, as well as a lack of a clear description on our part to point out that documents on the right-hand side are ranked. Refinery [11] provides a feature to easily filter subsets of documents from the output of LDA, which this case study corroborates.

Other feedback from these users:

- "LOVE, LOVE, LOVE the exported .csv!!"
- "Love how the topic names stay across the runs"
- "Love that it gives suggestions when you go to add a custom word"
- "We also liked how it shows you how many docs have a given word"

6. GENERAL DISCUSSION

Below, we synthesize what we learned from both case studies, and discuss points for future work.

6.1 Observations

Overall, our novel LDA workflow based on human-in-the-loop feedback was very positively embraced by our social scientists in both our case studies. Both sets of participants found that refining topics by confirming or rejecting individual words, and/or entire topics, helped to improve the quality and coherence of topics in subsequent iterations of LDA, by converting these choices into new topic-word distributions to serve as an informed initialization for a subsequent running of LDA.

In the evaluation of the GAO case study, we found a modest improvement in the coherence of topics based on the most salient *words* over the course of multiple iterations, and a significant improvement in the coherence of the most representative *documents* across topics found to be coherent.

We found it interesting that the quality of documents associated with a topic/theme improved more significantly than the semantic cohesion of words. We think part of this difference is due to Distill-ery's ability to automatically merge topics with identical labels, which likely helps the model discover more representative documents within a single semantic topic/theme. Further work should be done to make it even easier for users to both merge and split topics at will.

Another key takeaway from both case studies is that the automation of all aspects of running LDA, including uploading text data, tokenization and other pre-processing, and running the model itself, need to be baked into any tool that will be used by social scientists. After we developed our java web API enabled backend, all of our users were able to easily use Distill-ery with minimal training assistance, successfully uploading their documents, defining phrases important to their domain, producing LDA output, as well as refining topics found at each iteration by a combination of confirming, rejecting, and adding custom words from the vocabulary.

In both case studies, our participants found topic-word context to be very important in making sense of topics, and found that words alone were not enough to always make sense of the topics. In addition, both case studies found highlighting the specific word in a topic in context of the documents was also extremely useful to interpretation. This is consistent with similar findings by Lee et al [4], which presented evidence of many 'aha' moments from participants when provided additional context.

Both case studies found the topic labeling feature absolutely critical to improving interpretation of the topics. The topic labels allowed our users to quickly recall the overall semantic theme of the topic without expending further effort. Additionally, providing a label prediction algorithm between iterations helped to ground our users by connecting a subsequent iteration of LDA with the custom tagging work done from the previous iteration.

Our participants in both case studies found it difficult to know when they should be considered "done" iterating, and found it unclear how that point is determined. One of our case studies found that the intertopic distance map may be of potential value in this regard, by looking for the separation of bubbles into more distinct areas of the graph over the course of several iterations. However, further work needs to be done to research other possible ways to assist social scientists with this problem.

We found different patterns of refinement among the case studies, which was unexpected, but very welcome. Most interesting was the desire by our polticial science participants to want to create their own topic seeds based on individual intuitions about what they *wish* to see from the output of LDA, then allowing LDA to confirm/reject these intuitions. This desire fits very well with our iterative approach to a

workflow.

Pre-processing options, such as tokenization techniques like phrasing, are very important to social scientists. Social scientists do not necessarily understand what tokenization is, but certainly are quick to point out instances when individual words should instead be considered a semantic unit. In both our case studies, our users repeatedly requested the ability to customize how the tool creates phrases, as LDA has trouble correctly distinguishing from words alone, such as the case between as "United States" and "United Nations".

Overall, our users were fairly happy with the generated phrases from Justeson and Katz's regular expression. That said, it still missed some important phrases on its own. For those phrases, we give the user the possibility of uploading a custom set of phrases.

6.2 Future Work

Based on our initial results, there are many opportunities for future exploration. In this experiment, our assignment of the altered document-topic probability mass was very basic: good words received 80%, and all remaining words received the remaining 20% probability mass for each topic (minus the global stopwords, which received 0%). Further exploration should be done to explore how different probability mass functions might change the quality of the resultant topics. Existing topic refinement models, such as those proposed by Boyd-Graber et al. [9] could become extremely valuable to incorporate with our visual editor and process flow.

We need to give the user additional options for fine-tuning the running of LDA, such as number of iterations, etc. While these options are not likely to be used by every social scientist, it's important that we provide those extra knobs and dials to those who would use them if available.

Additional research needs to be conducted to design smarter algorithms for producing more relevant phrases, or developing modified versions of LDA that incorporate word sense. Improving techniques in these regards should help produce more coherent topics.

We need to provide a feature that keeps a complete log summary of the entire human-in-the-loop process. As our user in Case Study #2 stated, political science researchers need to be able to defend their work, and providing a log file will help increase transparency and reduce skepticism of these less-understood methods of content analysis.

We should explore the possibility of maintaining the Dirichlet distributions of incoherent topics across iterations rather than resetting incoherent topics back to a random initialization at each iteration. This could help to further assist LDA produce more coherent themes.

Further work should be done to determine better ways to "pre-form" topics for use cases where prior intuition about the potential themes in a corpus already exists. This could have the potential to expand the utility of LDA to a much wider audience of social scientists.

7. CONCLUSIONS

This approach shows great promise. The initial results from our iterative topic modeling process provides strong evidence that leveraging human-in-the-loop feedback from initial LDA output for use as an informed initialization for future iterations of LDA makes resulting topics more useful to social scientists.

Our claim is strengthened by the fact that we found similar results despite conducting two distinct use case studies, and we found marked improvement despite using a real-world dataset considered small and limited by most standard definitions. This approach appears to work well in a general sense, outside of an optimal environment.

Organizations doing traditional content analysis should feel confident incorporating an iterative topic modeling methodology into their existing processes to increase their efficiency without sacrificing the rigor of their existing analytic processes and quality of their results.

Distill-ery combines a visual topic model editor with an automated LDA workflow which hides the complexities of running LDA that many existing tools suffer from. Because it is built with the social scientist in mind, our tool has the potential to greatly expand the use of topic modeling among those doing text-focused content analysis, including political scientists, journalists, and researchers.

8. ACKNOWLEDGMENTS

I'd like to thank Philip Resnik for all of his thoughtful advice and helpful insights which were absolutely essential to refining the ideas presented in this paper. I'd also like to thank Andrew Stavisky for all of his assistance in providing us with real world data from GAO, for his help in providing critical feedback, as well as his time in discussions of the existing GAO processes and areas where he felt an impact could be made. I'd like to thank Sarah Croco and Jessica Liu for their patience with our tool as it was being developed, as well as all of their amazing feedback and phenomenal questions they provided us during the entire process of this effort. Last, I'd like to thank Viet-An Nyugen for his help with running Segan.

APPENDIX

A. GAO TAXONOMY CODES

GAO Taxonomy Codes

GAO Code	Category
A	**Best Aspects of Job**
A1	Variety / Never bored / Constantly changing / Always something new
A2	Learning new things
A3	Like fixing things
A4	Freedom / Autonomy / self-sufficient / make own schedule / work independently
A5	Training
A6	Job stability
A7	Challenge / sense of accomplishment / criticality of the work
A8	Get to work outside
A9	Pay
A10	People / Camaraderie
A11	Favorable military influence
B	**Worst Aspects of Job**
B2	Poor communication / planning / HQ understanding of technicians' workload and requirements / lack of mgt. support
B3	Length of time to get fully certified
B4	Rotating schedule
B5	Increase in workload / more work, less time / people
B6	Pay / Different pay systems depending on when started at FAA
B7	Lack of career progression
B8	Physical hazards âĂŞ need to work regardless of conditions
B9	Inability to fix some problems
B10	Equipment Issues - Age & Quantity of Equipment
B11	Impact on Family / Quality of Life
B12	Administrative tasks - GovTrip, Purchasing, TechNet, SALS
B13	Contract issues
C	**General / Misc. Staffing and Workload Concerns**
C1	Lack of staffing & other resources
C2	Poor staff planning
C3	Length of time to get fully certified (including delays in certification post-Academy/at work-site)
C4	Pay
C5	Low morale among Technicians
C6	FAA Maintenance Philosophy
C7	Attitudes toward Contractors
C8	Attitudes toward NextGen

D	Training Issues
D1	General lack of overall training on systems
D2	OJT / CBT
D3	Lack of availability of equipment to train on âĂŞ on site
D4	Some systems need on-site certification, not OK City
D5	New hires only being trained to changes/modifications, not original systems
D6	Lack of refresher training
D7	Academy Training
D8	Vendor provided training
D9	Feedback on Training
D10	Academy Issues
E	**Recommendations to Improve Staffing**
E1	Hire more people
E2	Focus on retention
E3	Hiring qualified personnel
E4	Make pay scale same for new hires (as was for older hires)
E5	Stop reliability-centered maintenance
E6	Treat technicians as equals to controllers
E7	Improve Benefits / Retirement system
E8	Improve management / staff relations (including communication, NextGen education and involvement)
F	**Recommendations to Improve Training**
F1	Familiarity training on systems before going to Academy
F2	Improve OJT / CBI
F3	Refresher training
F4	Improve Academy instructors
F5	Improve quality / depth of training
F6	Tailor training curriculum to ability/background and job function
F7	Improve annual training plan / Improve process for obtaining training
F8	Timely Academy training plus timely OJT with site/system familizarization is optimal method
F9	Consider innovative training changes, Tiger Teams at work sites
G	**System Examples**
G1-63	*63 examples of systems are part of this taxonomy, not reproduced here*

B. REFERENCES

[1] American presidency project. http://americanpresidency.org, 1999-2016. Accessed: 2016-04-20.

[2] C. W. Arnold, A. Oh, S. Chen, and W. Speier. Evaluating topic model interpretability from a primary care physician perspective. *Computer methods and programs in biomedicine*, 2015.

[3] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *the Journal of machine Learning research*, 3:993–1022, 2003.

[4] J. Boyd-Graber, T. Y. Lee, A. Smith, K. Seppi, N. Elmqvist, and L. Findlater. The human touch: How non-expert users perceive, interpret, and fix topic models. 2016.

[5] S. Deerwester, S. T. Dumais, G. W. Furnas, T. K. Landauer, and R. Harshman. Indexing by latent semantic analysis. *Journal of the American society for information science*, 41(6):391, 1990.

[6] J. M. Dickey. Multiple hypergeometric functions: Probabilistic interpretations and statistical uses. *Journal of the American Statistical Association*, 78(383):628–637, 1983.

[7] S. Geman and D. Geman. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, (6):721–741, 1984.

[8] J. Grimmer and B. M. Stewart. Text as data: The promise and pitfalls of automatic content analysis methods for political texts. *Political Analysis*, page mps028, 2013.

[9] Y. Hu, J. Boyd-Graber, and B. Satinoff. Interactive topic modeling. In *Association for Computational Linguistics*, 2011.

[10] J. S. Justeson and S. M. Katz. Technical terminology: some linguistic properties and an algorithm for identification in text. *Natural language engineering*, 1(01):9–27, 1995.

[11] D. Kim. Refinery. http://daeilkim.com/refinery.html, 2015. Accessed: 2015-12-11.

[12] C. D. Manning, M. Surdeanu, J. Bauer, J. R. Finkel, S. Bethard, and D. McClosky. The stanford corenlp natural language processing toolkit. 2014.

[13] A. K. McCallum. Mallet: A machine learning for language toolkit. http://mallet.cs.umass.edu, 2002.

[14] V. Nyugen and C. Musialek. Segan. http://github.com/cmoose/segan, 2015. Accessed: 2015-12-11.

[15] K. M. Quinn, B. L. Monroe, M. Colaresi, M. H. Crespin, and D. R. Radev. How to analyze political attention with minimal assumptions and costs. *American Journal of Political Science*, 54(1):209–228, 2010.

[16] J. Roberts. The echo chamber. www.reuters.com/investigates/special-report/scotus/#sidebar-analysis, 2015. Accessed: 2016-04-18.

[17] B. Shneiderman and C. Plaisant. Strategies for evaluating information visualization tools: multi-dimensional in-depth long-term case studies. In *Proceedings of the 2006 AVI workshop on BEyond time and errors: novel evaluation methods for information visualization*, pages 1–7. ACM, 2006.

[18] C. Sievert. Ldavis. http://cpsievert.github.io/LDAvis/reviews/vis/, 2016. Accessed: 2016-04-20.

[19] U.S. Government Accountability Office. FEDERAL AVIATION ADMINISTRATION: Agency is taking steps to plan for and train its technician workforce, but a more strategic approach is warranted. http://www.gao.gov/products/GAO-11-91, 2010. Accessed: 2015-12-19.