
Downloaded from [http://umiacs.umd.edu/~jbg/docs/2020_iui_control.pdf](http://umiacs.umd.edu/~jbg/docs/2020_iui_control.pdf)

Contact Jordan Boyd-Graber (jbg@boydgraber.org) for questions about this paper.
Digging into User Control: Perceptions of Adherence and Instability in Transparent Models

Alison Smith-Renner
amsmit@cs.umd.edu
University of Maryland
College Park, Maryland, USA

Varun Kumar
varunk@cs.umd.edu
Amazon Alexa
Cambridge, MA, USA

Jordan Boyd-Graber†
jbg@umiacs.umd.edu
University of Maryland
College Park, Maryland, USA

Kevin Seppi
kseppi@byu.edu
Brigham Young University
Provo, Utah, USA

Leah Findlater
leahkf@uw.edu
University of Washington
Seattle, Washington, USA

ABSTRACT

We explore predictability and control in interactive systems where controls are easy to validate. Human-in-the-loop techniques allow users to guide unsupervised algorithms by exposing and supporting interaction with underlying model representations, increasing transparency and promising fine-grained control. However, these models must balance user input and the underlying data, meaning they sometimes update slowly, poorly, or unpredictably—either by not incorporating user input as expected (adherence) or by making other unexpected changes (instability). While prior work exposes model internals and supports user feedback, less attention has been paid to users’ reactions when transparent models limit control. Focusing on interactive topic models, we explore user perceptions of control using a study where 100 participants organize documents with one of three distinct topic modeling approaches. These approaches incorporate input differently, resulting in varied adherence, stability, update speeds, and model quality. Participants disliked slow updates most, followed by lack of adherence. Instability was polarizing: some participants liked it when it surfaced interesting information, while others did not. Across modeling approaches, participants differed only in whether they noticed adherence.

CCS CONCEPTS

• Human-centered computing → Interactive systems and tools; Empirical studies in HCI; • Computing methodologies → Topic modeling.

KEYWORDS

Interactive machine learning, topic modeling, control, transparency, intelligent user interface evaluation

† Now at Google Research Zürich.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

IUI ’20, March 17–20, 2020, Cagliari, Italy
© 2020 Association for Computing Machinery.
ACM ISBN 978-1-4503-7118-6/20/03...$15.00
https://doi.org/10.1145/3377325.3377491

1 INTRODUCTION

Machine Learning (ML) is common in today’s data-rich society. These techniques build models of data, either with explicit training labels or by finding patterns when labels are not provided. However, they are not perfect: data are noisy, models are deficient, and humans’ needs and understanding sometimes conflict with ML output [4]. In these cases, a human–machine collaboration is required to iteratively improve and adapt models. Users can control models by providing input such as additional training labels, re-weighting features, or modifying the underlying data representation.

Effective human-machine collaboration requires model transparency: users who understand models better can also better correct models’ mistakes [29, 45]. However, increased transparency has another effect in interactive ML (iML): when users provide input to the model, they can tell whether the model uses their feedback predictably (or not). For example, suppose users re-weight a regression model’s features for predicting property prices, specifying that house color should not influence future predictions; they will be rightly surprised if the model later explains a predicted price using the color. Therefore, with transparent models—where controls are easy to validate—we cannot provide users with mechanisms to control the model and expect for a positive outcome—we must also consider how models update and what cascading side effects might occur. This need introduces a problematic tension: models must balance the user requested changes with faithfully modeling data.

This paper explores how users perceive and are affected by two specific aspects of control as it relates to predictability: whether input is applied as expected (adherence) and whether other unexpected changes occur (instability). iML models also vary in other attributes that affect user experience, particularly their latency, or how long they take to update, and their performance, or how well they model the data or accuracy on held out test sets. These are not just attributes of iML systems; human–computer interaction guidelines prescribe that interactive systems should be predictable,
controllable, and provide immediate updates [22, 47]. While prior work has highlighted control and predictability for intelligent systems [25], the interaction between these constructs has not been fully explored, particularly in transparent models, where they are more easily perceived.

We study adherence, instability, latency, and quality in Human-in-the-Loop Topic Modeling [5, hl-tm]. In hl-tm, users are exposed to and interact directly with model representations, meaning users can more easily validate when their changes are not applied as expected—or if other unexpected changes also occur compared to less transparent systems with abstract representations. Prior studies have exposed adherence and instability of hl-tm through user interviews [35, 50]. However, these attributes and their effects on end users have not been studied at a large scale or compared between models.

To explore how participants responded to varied system adherence, instability, latency, and quality, 100 crowdworkers used three hl-tm systems for document understanding and organization. Rather than artificially manipulating adherence, instability, latency, and quality, we chose to compare three distinct modeling approaches that vary in these characteristics as a result of using different objective functions and optimization strategies. We explore whether user perceptions and experience differ between these systems, and examine user behavior more generally, to better understand how end users approach and interact with interactive models with easy-to-validate controls.

The systems had significantly different computed adherence, instability, and latency, yet only perceived adherence differences were significant. This finding suggests that participants noticed when the systems did not apply their input as expected, a phenomenon that was particularly evident for easy-to-track refinements, such as adding or removing topic words. Participants were polarized by instability: some liked that it surfaced interesting information, others disliked it, and even others reported not noticing it. Participants thought all three systems were slow but performed well. Interestingly, the majority of participants thought that they improved the model quality, but on average they reduced computed topic coherence, suggesting a possible disconnect between traditional topic coherence measures and user perceptions of hl-tm quality. Overall, users trusted all three systems and thought the task was easy, yet some were frustrated, particularly by slow updates.

This paper provides three major contributions to our understanding of user interaction with transparent systems: (1) an analysis of how users perceive adherence and instability, and whether these attributes affect users’ experience; (2) an understanding of the trade-offs between system attributes of adherence, instability, latency, and performance; and (3) design recommendations for transparent, interactive systems.

2 BACKGROUND

We review control and transparency in ml and provide background on hl-tm, the case we use to explore these attributes.

2.1 Control with Transparent Ml

End users want to understand how ml models work [37]. Models can provide transparency through explanations or justifications for particular decisions or actions [8, 9]. Transparent models might also expose their inner workings, or how they model the underlying data for a deeper understanding of how they operate [14]. For example, Simonyan et al. [49] increase the transparency of deep Convolutional Networks by producing artificial images representative of learned image classes. As transparency increases, end users form better mental models, which in turn increases end user trust, satisfaction, and leads to continued usage [20, 30, 36, 44].

End users separately want and need mechanisms for control, both for user interfaces broadly [48] and for ml-based systems [3]. Specifically, allowing users to control models can manage user expectations [28] and increase satisfaction [46, 53]. Transparency is particularly important when users are given control [29, 45], as making users aware of how models work in turn makes them better at providing feedback. However, increased transparency also means that users can better discern what models do with their feedback, or whether models incorporate it predictably. For opaque systems, providing "difficult-to-validate" controls, whether or not they work, can increase satisfaction [53]. But how will users react to unpredictable behavior when controls are easier to validate? This paper explores two specific aspects of control as it relates to predictability: adherence—how well models apply user specifications during updates—and instability—whether models make any other changes.

2.2 Topic Modeling with a Human-in-the-Loop

We explore adherence and instability in Human-in-the-Loop Topic Modeling (hl-tm). Statistical topic models automatically identify the themes or topics that occur in collections of documents [11], and are typically represented as collections of topics, where topics are represented as their top words and associated documents [14]. Topic models allow users to understand and explore document collections by the themes they discuss.

Latent Dirichlet Allocation [10, lda] is a common unsupervised topic modeling algorithm, which models each document in the corpus as a distribution of topics and each topic as distribution of words in the vocabulary. However, topic models are not always perfect [12]. Several extensions to lda incorporate human knowledge to improve topic models [26, 27, 42, 43, 54, 55, 57, hl-tm]. With such techniques, users specify model refinements, such as words or documents to be removed from topics.

Our hl-tm approach is transparent in that users are exposed to and interact directly with the underlying model—topic words (θ) and associated documents (φ)—as opposed to abstract representations such as labeled folders. Therefore, it is a good case for exploring adherence and instability: users can more easily track if their changes are applied as expected (or other unexpected changes occur in the model). These issues may be less obvious in less transparent systems, such as recommenders [21], where users interact with abstract representations (recommended items) instead of the underlying model (decomposed user-item interaction matrix). hl-tm is also a representative document understanding system, particularly one where users focus on both words and documents—more complex than interactive clustering [16], for example.

Like other ml models, hl-tm techniques differ how they adhere to input and whether they make any other unexpected changes.
Prior studies have exposed these attributes through user interviews [35, 50], yet the effects of these attributes on users have not been fully explored, either with many users or comparatively. Kumar et al. [31] implemented a set of user-preferred refinements [35] using three different modeling approaches and measured adherence provided by the different approaches using simulations. However, their simulated user experiments are prima facie implausible: they ignore human variability, the depth of human insight, and the reaction of humans to imperfect model updates. To correct these oversights, we use these three approaches to explore how users perceive adherence and instability and whether they affect user experience and behavior.

Instability is not a new concept in ML; deterministic algorithms are stable—they always produce the same output given the same input. Prior work in statistical topic modeling explores instability between learned topic models on different runs [7, 18]. This paper specifically explores whether users perceive such instability on model updates.

3 COMPARATIVE EVALUATION OF HL-TM MODELING APPROACHES

For this study, crowd workers interacted with a topic model to organize documents using one of three contrasting HL-TM approaches based on LDA [10]. The approaches support the same set of nine refinement operations (e.g., merging topics and removing words or documents from topics), and differed only in implementation details, as these criteria affect model attributes, such as adherence, instability, quality, and latency.

This study used a between-subjects experimental design with a single factor (Modeling Approach): informed priors using Gibbs sampling (info-gibbs), informed priors using variational inference (info-vb), and constraints using Gibbs sampling (const-gibbs).

The goal of this study was to explore how users perceive and interact with transparent systems with varied attributes: adherence, instability, latency, and quality. This study explored specifically:

(RQ1) How do users perceive instability and adherence across the three HL-TM approaches? (RQ2) How does user experience vary given these differing attributes? (RQ3) How do users behave with the three HL-TM approaches?

3.1 Modeling Approaches

We implemented three HL-TM systems, based on LDA, following modeling approaches proposed in prior work [31]. These modeling approaches differ in how user input (e.g., added words) is conveyed to the model—informed priors [50] or constraints [56]—and inference strategies—variational inference [10] or Gibbs sampling [19].

While other topic modeling approaches exist [24, 33], we chose these LDA-based variants because they support the same user-preferred refinement set. For example, “anchor words”-variants [38] also generate topics, but cannot support word-level operations like adding words. Also, these approaches may differ by the attributes of interest. For example, prior work asserts that informed priors better adhere to refinement operations [31], and Gibbs sampling-based methods can yield more coherent topics [41]. Also, Gibbs sampling and variational inference have different convergence rates [6]. While Gibbs sampling is often preferred for small datasets and interactive settings because of its low latency, variational inference can scale to millions of documents [23, 58]. Our setting allows a focused, task-center comparison (Section 3.9.1).

For info-gibbs and const-gibbs, we trained initial LDA models with 300 Gibbs sampling iterations and default Mallet toolkit1 hyperparameters (α = 0.1; β = 0.01) and, for info-vb, 30 EM iterations. For each subsequent update during the task, we applied the refinement and ran inference.

3.2 Refinement Implementations

For each of the three HL-TM modeling approaches, we implemented the same nine refinement operations preferred by users in prior work [35, 40, 50]. These include four topic-level refinements: add word, change word order, remove word, remove document and five model-level refinements: merge topics, split topic, create topic, delete topic, add to stop words.

For remove word, add word, remove document, merge topics, split topic, change word order and create topic, we applied the refinements following the implementation proposed by Kumar et al. [31]. For add to stop words, to add a word w to the stop words list, we excluded w from the model vocabulary. For delete topic, to delete a specified topic t, in all three models, we first forgot all latent topic assignment which were assigned to t, and then reduced the number of topics by one.

After refinements were applied, we ran inference for N iterations to limit latency (rather than running inference until convergence). Moreover, all refinements have different levels of complexity, meaning the models converge faster for certain refinements than others. For example, add to stop words is a simpler refinement than create topic, and hence requires fewer iterations to converge. For each refinement, we empirically fine-tuned N on 9000 tweets randomly selected from a different dataset.2 In particular, to fine-tune N for a refinement, we randomly applied a refinement multiple times and observed how fast the model converged. For info-gibbs and const-gibbs, N ranged from one for add to stop words to 20 for create topic. For info-vb, N varied from one for add to stop words to four for create topic.

3.3 Dataset

For the study we used the Twitter Airline Sentiment Dataset, which includes tweets directed at various common airlines (e.g., United, Southwest Airlines, Jet Blue) and manually tagged by sentiment (positive, negative, neutral).3 We produced initial topic models of 10 topics from only the 9,178 negative sentiment tweets, as these reflect a distinct set of complaints regarding air travel.

3.4 Task Interface

The HL-TM task user interface was the same for all three modeling approaches (Figure 1). The topics are listed on the left, each initially represented by a generic topic label (e.g., “Topic 1”) and the three most probable words for the topic. The selected topic is on the right, which displays the top 20 topic words and the top 20 topic

1http://mallet.cs.umass.edu/
2https://www.kaggle.com/kazanova/sentiment140/
3https://www.kaggle.com/crowdflower/twitter-airline-sentiment
documents. Documents are ordered by their probability for the topic $t$ given the document $d$, or $p(t|d)$. Each word, $w$, is ordered and sized by its probability for the topic $t$, or $p(w|t)$; this simple word list representation provides a quick understanding of the topic [1, 51]. Hovering or clicking on topic words highlights the word in the displayed document snippets. Participants can click the pencil icon to rename the topic labels to be more descriptive.

Participants can explore and update the model using the set of nine refinement operations: click “x” next to words or documents to remove them, select and drag words to re-order them, type new words into the input box and press “enter” to add them, select a word and click “remove selected word from all topics” to add it to the stop words list, click “delete topic” to remove the selected topic, or click “create a new topic”, “split”, or “merge” (in the topic list) to enter into create, split, or merge modes, respectively (Figure 1). Each refinement is immediately saved and the model is updated. After updates, participants can undo to revert models to prior states.

3.5 Participants

We recruited 100 participants (32 male and 68 female) on the Upwork platform. Participants were required to have a 90% or higher job success score and be native or bilingual English speakers. We designed the task to take approximately 60 minutes and paid participants 20 USD. We used Upwork instead of other common crowdworker platforms (e.g., Mechanical Turk), to recruit more motivated participants; participants were paid a higher rate and could always contact one of the researchers in case of questions.


To understand participants’ prior exposure to topic models and machine learning, as this could affect our results, study participants rated prior experience with statistical topic modeling and

---

4https://www.upwork.com/
machine learning, respectively. Participants varied for prior experience (rated on a scale from one to five) with topic models ("none" (one): 44, two: 25, three: 18, four: seven, "significant" (five): six) and machine learning ("none" (one): 44, two: 19, three: 19, four: nine, "significant" (five): seven).

3.6 Procedure

Each participant was randomly assigned to one of the three modeling approaches and all used the same %TM user interface (Figure 1). Each user got a unique starting model from a pool of 50 pre-trained initial %TM models with 10 topics for each of the three %TM modeling approaches. Given the assigned approach, we randomly selected an initial topic model from the pool of pre-trained models and then removed the selected model from the pool. The study began with a tutorial, which introduced participants to topic modeling, relevant terminology, and the task interface. The tutorial also required participants to experiment with each of the nine refinement operations. After the tutorial, participants were given the following task instructions:

”Imagine you have been asked to write a travel blog post about the common complaints that travelers have when flying. The system has gathered 9000 tweets of people complaining about their air travel experience directed at various popular airlines and has generated an initial set of 10 topics to organize these air travel complaint tweets. Use the tool to improve these topics, so that you can write a blog post about common air travel complaints with a few example tweets from each. You do not need to write the actual blog post as part of this task.”

Participants were then asked to spend 30 minutes interacting with the model and to click the “finish task” button when they were happy with the organization they had achieved. The interface required participants to spend at least 20 and no more than 45 minutes on the task. The task goal and time elapsed were denoted in the interface (Figure 1).

After the task, participants completed a survey containing closed- and open-ended questions on their perceptions and experience with the system (Table 1) and which refinements they felt were the most and least useful, with follow up “why” questions. Participants also responded to whether they noticed any unexpected behavior while using the tool and what they liked and did not like about using the tool for the task.

3.7 Measures

We report on nine overall subjective measures, collected using seven-point rating scales (Table 1): four user experience measures (frustration, trust, task ease, confidence) and five user perception measures (perceived adherence, perceived instability, perceived latency, final model satisfaction, and perceived improvement). We also report on subjective per-refinement adherence, collected using seven-point rating scales (strongly disagree to strongly agree) for nine statements of the form, “the system incorporated the [refinement] operation as I asked it to.” These statements also included a "did not use operation" option.

<table>
<thead>
<tr>
<th>Measure</th>
<th>Statement</th>
</tr>
</thead>
<tbody>
<tr>
<td>frustration</td>
<td>&quot;Using this tool to perform the task was frustrating&quot;</td>
</tr>
<tr>
<td>trust</td>
<td>&quot;I trusted that the tool would update the organization of the tweets well&quot;</td>
</tr>
<tr>
<td>task ease</td>
<td>&quot;It was easy to use this tool to perform the task&quot;</td>
</tr>
<tr>
<td>confidence</td>
<td>&quot;I was confident in my specified changes to the tool&quot;</td>
</tr>
<tr>
<td>final model satisfaction</td>
<td>&quot;How satisfied are you with the final organization of the tweets?&quot;</td>
</tr>
<tr>
<td>model improvement</td>
<td>&quot;How do you think the final organization compares to the initial organization of the tweets?&quot;</td>
</tr>
<tr>
<td>low latency</td>
<td>&quot;It was easy to use this tool to perform the task&quot;</td>
</tr>
<tr>
<td>adherence (overall)</td>
<td>&quot;How do you think the final organization compares to the initial organization of the tweets?&quot;</td>
</tr>
<tr>
<td>instability</td>
<td>&quot;The tool made unexpected changes beyond what I asked it to make&quot;</td>
</tr>
</tbody>
</table>

We also report on quantitative measures of the system attributes: adherence, instability, latency, and quality (initial, final, and improved). To compute adherence for each of the nine refinements we use the metrics provided by Kumar et al. [31]:

- **add word, remove word, and change word order**: treat the topic as a ranked word list, and then take the ratio of the actual rank change (where the added, removed, or reordered word is in the updated model) and the expected rank change.
- **remove document**: compute similarly to remove word, except treat the topic as a ranked document list.
- **create topic**: compute the ratio of the number of seed words in the created topic out of the total number provided.
- **split topic**: compute the average adherence of the parent and child topic, using the adherence measure for create topic.
- **merge topics**: compute the ratio of the number of the words in the merged topic that came from either of the parent topics over the total number of words shown to the user.
- **add to stop words and delete topic**: these refinements are deterministic, and therefore always have a perfect adherence score.

Adherence is measured on a range from 0.0, meaning the system ignores the user’s input, to 1.0, meaning the system does exactly as the user asks. The exception is adherence to change word order, which ranged from $-\infty$ to $\infty$, and where a negative adherence value means the system did the opposite of what the user asked. For example, if a user moves a word up two positions, but it is instead moved down one, the adherence would be $-5$. Overall adherence is computed as the average adherence score over all refinements applied by the user.

To estimate the instability caused by a refinement, we use a modified topic-term stability metric [7]. We first compute the difference between each topic as 1.0 minus the overlap coefficient [39] between the top 20 words of the topic, and then take the ratio. Instability is then measured as the average difference between each topic excluding the refined topic(s). Put simply, we compute what percentage of topic words are removed after an update for the untouched topics. Instability is scored from 0.0 (all topics the same) to 1.0 (all topics completely different). Latency is the time the model...
Table 2: Measures for system attributes: instability, adherence, latency (seconds), and quality—final model quality (coherence) and percent improvement. Coherence scores multiplied by 1000 for readability. Responses reported as “mean, σ”. Kruskal-Wallis results reported as χ²(2), p*. The modeling approaches differed significantly (bold) for all computed attributes except improvement; cell shading for significantly differences highlights better approaches (darker is better).

<table>
<thead>
<tr>
<th>Attribute</th>
<th>info-gibbs</th>
<th>const-gibbs</th>
<th>info-vb</th>
<th>Kruskal-Wallis</th>
</tr>
</thead>
<tbody>
<tr>
<td>Adherence</td>
<td>.841</td>
<td>.701</td>
<td>.829</td>
<td>20.8, p&lt;.001</td>
</tr>
<tr>
<td>Stability</td>
<td>.12.03</td>
<td>.12.03</td>
<td>.03.03</td>
<td>1754.8, p&lt;.001</td>
</tr>
<tr>
<td>Latency (s)</td>
<td>15.2, 6.2</td>
<td>19.3, 9.2</td>
<td>20.4, 5.9</td>
<td>18.1, p&lt;.001</td>
</tr>
<tr>
<td>Final Quality</td>
<td>7.4, 3.5</td>
<td>7.0, 1.9</td>
<td>5.7, 1.5</td>
<td>8.5, .014</td>
</tr>
<tr>
<td>Improvement</td>
<td>6%, 42%</td>
<td>4%, 34%</td>
<td>-7%, 30%</td>
<td>1.4, .489</td>
</tr>
</tbody>
</table>

We disqualified five of the 100 participants because they made an outlying number of survey response “mistakes” on per-refinement adherence statements. We considered a response to be a “mistake” if the participant said they had used a refinement for the task when they had not, or vice versa, and used an interquartile range (IQR) approach to determine outliers based on the count of mistakes [52]: the median number of mistakes was two, and the upper quartile bound for outliers (Q3 + 1.5IQR) was five (out of nine possible mistakes). Removing outliers above this bound resulted in 95 participants in our final dataset: 31 in the info-gibbs condition, 33 in the const-gibbs condition, and 31 in the info-vb condition.

For quantitative analysis, we used separate Kruskal Wallis tests to determine significance across the conditions for each of the subjective rating responses and the quantitative attributes. For qualitative analysis, we followed a thematic approach [15], and coded the open-ended responses related to what participants found unexpected, liked and did not like, and which refinements they found were most and least useful. Two annotators independently coded a random subset of 20 of the 95 responses for each of the statements regarding what was unexpected, what participants liked, and what they disliked; agreement was scored using Cohen’s κ: κ = .93 for unexpected responses, κ = .88 for liked responses, and κ = .89 for disliked responses.

3.9 Results

Each of the 95 participants started with a distinct initial random topic model and applied refinements with the goal of improving the model for their imagined travel blog. In the following sections, we provide detailed results regarding computed model attributes followed by user perceptions, experience and behavior given these different attributes, and with interactive topic models in general. We refer to participants throughout this section as P1-P95.

3.9.1 Computed Differences. The three modeling approaches differed significantly for four out of the five computed attributes: adherence, instability, latency, and final model quality but not model improvement (Table 2). The Gibbs sampling approaches (const-gibbs and info-gibbs) had higher final model quality than variational inference (info-vb), while variational inference was more stable than Gibbs. Informed priors with Gibbs sampling (info-gibbs) provided the fastest updates over const-gibbs and info-vb. Finally, informed priors (info-gibbs and info-vb) provided higher control than constraints (const-gibbs).

Analyzing adherence in more detail. Table 3 shows the average computed per-refinement adherence for each modeling approach. Computed adherence differed significantly across modeling approaches for four of the nine refinements: const-gibbs provided less control for add word, change word order, and create topic than the other approaches. For split topic, info-vb provided the most control followed by const-gibbs, and info-gibbs provided the least.

3.9.2 User Perceptions. We analyzed participants’ perceptions regarding adherence, instability, latency, and model quality through subjective responses (Figure 2 and Figure 3). While computed adherence, instability, latency, and final model quality differ across modeling approaches, for subjective measures, only adherence is significantly impacted by condition: participants in const-gibbs perceived lower adherence than the other modeling approaches. It is important to note that we did not control for these characteristics nor for the magnitude of their differences, which may explain why users did not perceive differences in all dimensions.

Table 3: Computed per-refinement adherence measurements reported as “mean, σ”. Kruskal-Wallis results reported as χ²(2), p*. There were significant differences (bold) between modeling approaches for add word, change word order, create topic, and split topic; cell shading reflects adherence to that refinement (darker is better).

<table>
<thead>
<tr>
<th>Refinement</th>
<th>info-gibbs</th>
<th>const-gibbs</th>
<th>info-vb</th>
<th>Kruskal-Wallis</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add word</td>
<td>.99, .01</td>
<td>.62, .28</td>
<td>.06, .04</td>
<td>49.4, p&lt;.001</td>
</tr>
<tr>
<td>Remove word</td>
<td>.91, .17</td>
<td>.97, .08</td>
<td>.99, .03</td>
<td>3.4, .180</td>
</tr>
<tr>
<td>Remove doc</td>
<td>.78, .32</td>
<td>.88, .22</td>
<td>.69, .28</td>
<td>3.6, .160</td>
</tr>
<tr>
<td>Change order</td>
<td>.67, .26</td>
<td>.86, .50</td>
<td>.53, .36</td>
<td>.29,7, p&lt;.001</td>
</tr>
<tr>
<td>Create topic</td>
<td>1.0, 0</td>
<td>.53, .24</td>
<td>1.0, 0</td>
<td>21.9, p&lt;.001</td>
</tr>
<tr>
<td>Delete topic</td>
<td>1.0, 0</td>
<td>1.0, 0</td>
<td>1.0, 0</td>
<td>NA</td>
</tr>
<tr>
<td>Merge topics</td>
<td>.82, .08</td>
<td>.79, .07</td>
<td>.83, .09</td>
<td>4.0, .130</td>
</tr>
<tr>
<td>Stop word</td>
<td>1.0, 0</td>
<td>1.0, 0</td>
<td>1.0, 0</td>
<td>NA</td>
</tr>
<tr>
<td>Split topic</td>
<td>.80, .27</td>
<td>.88, .08</td>
<td>.94, .12</td>
<td>10.6, .007</td>
</tr>
</tbody>
</table>
Overall, participants thought the systems adhered to their input \((M = 5.3 \pm 1.8)\), but were mixed on whether the systems were unstable \((M = 3.3, \sigma = 2.2)\). Participants thought the final models improved \((M = 5.8, \sigma = 1.1)\) and they were satisfied with the quality \((M = 5.1, \sigma = 1.3)\), but they thought the model updates were slow \((M = 2.7, \sigma = 1.6)\).

Participants noticed when word-level refinements did not adhere. Adherence was lower for const-gibbs than other approaches \((M = 5.5, \sigma = 1.3)\) than const-gibbs \((M = 4.6, \sigma = 1.9)\). These differences were significant \((\chi^2(2) = 6.3, p = .042)\).

Perceived adherence was also significantly lower for const-gibbs for two easy-to-validate word-level refinements \((M = 5.5, \sigma = 1.3)\) than const-gibbs \((M = 4.6, \sigma = 1.9)\). These differences were significant \((\chi^2(2) = 10.1, p = .006)\) and change word order \((\chi^2(2) = 11.5, p = .003)\). However, there was no significant difference between the modeling approaches for perceived adherence of the create topic \((\chi^2(2) = .9, p = .62)\) or split topic refinements \((\chi^2(2) = 3.6, p = .17)\), even though these differed for computed adherence \((\chi^2(2) = 9.0, p = .01)\). This is perhaps because it is harder for users to discern perfect refinements (all requested words appear in the new topic) from those that are “good enough”.

Participants were mixed on whether they observed instability. The computed instability metric shows that the info-vb condition was significantly more stable than the other modeling approaches \((M = 5.5, \sigma = 1.3)\) than const-gibbs \((M = 4.6, \sigma = 1.9)\). These differences were not significant \((\chi^2(2) = 5.6, p = .057)\).

Participants thought they improved the models, but coherence scores disagree. We measured quality and improvement using qualitative (Figure 3)—judged by the user—and quantitative—automatic topic coherence, Table 2—methods. Confirming that our initial random model creation was effective, there were no significant differences between modeling approaches for the initial model quality \((\chi^2(2) = 4.1, p = .130)\). Automatic coherence declined on average for models, most notably for info-vb, confirming previous reports that variational inference can produce less coherent topics than Gibbs sampling \((\sigma = 2.2)\). In contrast, participants believed they improved the models: while only 42% of the 95 participants improved the model \((M = 5.0, \sigma = 1.1)\), 98% thought the final model was better than the initial model \((M = 5.1, \sigma = 1.3)\) and const-gibbs \((M = 5.8, \sigma = 1.1)\), these differences were not significant \((\chi^2(2) = 5.6, p = .03)\).
words and associated documents. Additionally, ideal metrics would be less dependent on the data being modeled.

Participants thought all the systems were too slow. Objectively, the info-gibbs condition had significantly faster updates (Table 2). However, users thought all the systems were slow (Figure 2), and the perceived latency differences between modeling approaches were not significant ($\chi^2(2) = 1.0, \ p=.610$). This was likely a combination of participants wanting the systems to be faster and of unrealistic expectations for speed given participants’ experiences in the tutorial. For example, P71 (info-gibbs) asked, “is there any way to make it a bit faster?... It would be better if the tutorial wasn’t so fast... so you don’t have the expectation of speed with this tool.”

3.9.3 User Experience. To understand how variations in adherence, instability, latency, and model quality may affect user experience, participants responded to statements regarding frustration, trust, task ease, and confidence (Figure 4). Participants were confident, found the task easy, and trusted the tool: mean response for these measures across all modeling approaches was 5.4, 5.0, and 5.3 out of 7, respectively. Participants were neutral regarding frustration, at 3.5 out of 7 for all models, with info-gibbs the least frustrating ($M = 2.9, \ \sigma = 1.7$) and const-gibbs ($M = 3.8, \ \sigma = 1.8$) and info-vb ($M = 3.7, \ \sigma = 2.2$) the most. There were no significant effects of modeling approach on these experience measures, but the open-ended responses provide additional insight into how adherence, instability, and so on affect user experience.

Table 4: Likert scale responses for agreement with statements of the form “the system incorporated the [refinement] operation as I asked it to” for each of the nine refinements. Measurements reported as “mean, $\sigma$”. Kruskal-Wallis results reported as “$\chi^2(2)$, $p$”. Overall, change word order had low perceived adherence, and there were significant (bold) perceived adherence differences between modeling approaches for add word and change word order; cell shading reflects participant perception that the modeling approach adheres to that refinement (darker is better).

<table>
<thead>
<tr>
<th></th>
<th>info-gibbs</th>
<th>const-gibbs</th>
<th>info-vb</th>
<th>Kruskal-Wallis</th>
</tr>
</thead>
<tbody>
<tr>
<td>add word</td>
<td>6.1, 1.5</td>
<td>4.6, 2.5</td>
<td>6.5, 1.4</td>
<td>9.2, 010</td>
</tr>
<tr>
<td>remove word</td>
<td>6.5, 1.1</td>
<td>5.9, 2.1</td>
<td>6.7, 6</td>
<td>.8, 660</td>
</tr>
<tr>
<td>remove doc</td>
<td>6.3, 1.5</td>
<td>6.8, 5</td>
<td>5.6, 2.1</td>
<td>5.0, 080</td>
</tr>
<tr>
<td>change order</td>
<td>4.9, 2.2</td>
<td>2.9, 2.5</td>
<td>5.2, 2.4</td>
<td><strong>11.5, 003</strong></td>
</tr>
<tr>
<td>create topic</td>
<td>6.0, 1.9</td>
<td>6.1, 1.4</td>
<td>6.3, 2.1</td>
<td>9.620</td>
</tr>
<tr>
<td>delete topic</td>
<td>6.8, .7</td>
<td>6.4, 1.3</td>
<td>6.9, 3</td>
<td>1.5, 470</td>
</tr>
<tr>
<td>merge topics</td>
<td>6.7, .8</td>
<td>6.8, 5</td>
<td>6.7, 7</td>
<td>.2, 900</td>
</tr>
<tr>
<td>stop word</td>
<td>6.0, 2.0</td>
<td>6.3, 1.4</td>
<td>6.6, 7</td>
<td>.3, 860</td>
</tr>
<tr>
<td>split topic</td>
<td>5.6, 2.2</td>
<td>5.9, 2.0</td>
<td>6.9, 3</td>
<td>3.6, 170</td>
</tr>
</tbody>
</table>

Open-ended responses regarding likes, dislikes, and unexpected behavior. Our coding of open-ended responses (Section 3.8) resulted in seven disliked, seven liked, and five unexpected codes.

Participants disliked “latency” the most (42 of 95) followed by “lack of control” (21 participants). Ten participants thought the systems were “missing functionality”, requesting support for dragging documents between topics or comparing two topics at once. Eight participants thought the tool was “overwhelming”, while five said there was “nothing” they did not like. Five disliked “model qualities”, such as too many similar topics (P46, const-gibbs). Finally, two participants mentioned disliked “instability”.

Participants liked that the systems were “useful” for organizing and filtering the documents (40 of 95) and that they were “intuitive” (28). Ten participants liked the “refinements”, particularly when they worked as expected, such as P22 (const-gibbs), “the removing of terms was neat and operated as expected”, while three participants said they liked when the systems “worked as expected”. Five participants liked the systems’ “design”, two participants said they liked “instability”, and one liked that the tool was “fast”.

Of the measured attributes, participants thought “lack of control”, or adherence, (35 of 95) was most unexpected, such as P14 (info-gibbs) who said, “once the change word order did not happen, even though I tried it three times”, followed by “slowness” (22) and “instability” (12). Twenty participants said “nothing” was unexpected and six mentioned “other” things, like issues with the tutorial.

Instability was the most polarizing attribute. Not all noticed it, but those that did disagreed, confirming prior work [50]. While 12 of 95 participants said “instability” (as opposed to other attributes) was unexpected, some participants, such as P79 (info-vb) said, “I didn’t expect the word list to automatically update after adding a new word but I thought that was cool.” While other participants said instability was negative, such as, “I [removed a word] and saw it in a later topic...bad stu!” (P20, info-gibbs).
3.9.4 User Behavior. In addition to measuring participants’ subjective responses regarding whether they perceived differences in system attributes and how this affected their experience, we were also interested in understanding how users interact with these systems. On average, each participant used six ($\sigma = 1.4$) of the nine operations to make a total of 31.3 ($\sigma = 16.1$) changes to their model. In the following, we detail whether user behavior differed given the varied attributes and how users behaved with these systems.

Low adherence may have led participants to stop the task early. Table 5 shows the average time spent on the task and number of refinements for each condition. The const-gibbs modeling approach had significantly slower updates, so we might have expected those participants to spend the longest time on the task, but they did not: participants in the const-gibbs condition on average made fewer refinements ($M = 27$, $\sigma = 13$) and spent significantly less time on the task ($M = 1859$ seconds, $\sigma = 352$) than with the other modeling approaches. This might be explained by adherence: the const-gibbs modeling approach had significantly lower computed and perceived adherence (Table 2 and Figure 2), suggesting participants may have abandoned the task if they thought the system was ignoring their input.

Participants used “undo” infrequently, but reverted delete and split topic the most. Participants used “undo” 58 times to revert after applying a refinement. 36 of the 95 participants used “undo” an average of 1.6 times ($\text{min} = 1$, $\text{max} = 5$). Figure 5 shows the distribution of refinements that preceded undo normalized by the usage of the refinement. The most frequently undone refinements were delete topic, which was undone 10% of the time, and split topic, which was undone 8% of the time.

The high frequency of undoing delete topic is unexpected. While we had anticipated that participants might undo if operations were not applied as expected, all systems perfectly adhered to the delete topic refinement; that is, in these cases, participants were likely exhibiting experimentation behavior [2]—perhaps looking for instability to update other areas of the model and then undoing the change if they were not happy with it.

Participants attended to prominent and low quality topics. Figure 6 shows which topics were refined by participants based on their location in the topic list (left) and their relative coherence (right). All participants saw a random topic model with random topic ordering, yet participants focused their refinements on the topics at the top of the list ($\text{corr} = −0.98$) and on the topics that had the lowest coherence ($\text{corr} = −0.94$).

*Which refinement operations were used and preferred?* Participants refined models at the topic-level more often than at the model-level: remove document was used most (8.0 times per participant), followed by remove word (7.5), change word order (6.5), and add word (4.1). Of the topic-level refinements, the two least used (add word and change word order) were also those that had lower perceived adherence. The most common model-level refinement was merge topics, used 2.4 times per participant on average, followed by add to stop words (1.4), delete topic (0.7), split topic (0.5), and create topic (0.5). Participants specified which refinements were most and least useful: merge topics was overwhelmingly favored (46 of 95 participants said it was most useful), while change word order was unpopular (25 of 95 participants thought it least useful). To better understand why, we look to the open-ended questions.

**Participants may have disliked that change word order did not work as expected.** Thirteen of the 25 participants who thought change word order was the least useful were in the const-gibbs condition, likely because this refinement had significantly lower computed and perceived adherence than in other modeling approaches. Further, many of the participants who did not like change word order explained that it “did not work” or had no noticeable effect on the updated model. For example P98 (const-gibbs) said, “for some reason, [change word order] would not work with me.”

**Merge topic was a useful refinement for the data and task.** Of 95 participants, 49 said that merge topic was the most useful refinement, while none thought it least useful. Many of these participants thought merge topic was “especially useful for the task and model”; for example, P82 (const-gibbs) said, “there were multiple topics generated that meant the same thing as another. Putting them together made it more organized.”

4 DISCUSSION AND FUTURE WORK

This paper explores users’ perceptions, experience, and behavior with systems with easy-to-validate controls—in particular, those
that provide varied levels of control for both adherence and instability. This section discusses implications and design recommendations for such systems as well as limitations of this study and suggestions for future work.

Users want to be heard. End users want to be in control [28, 53], but what about when systems cannot respect user inputs? While users may expect that their input will be adhered to, as demonstrated by qualitative comments in our study, modeling approaches differ in how user input is incorporated, particularly when it conflicts with the underlying data. For example, suppose a user interacting with a property pricing tool tries to remove all weight from crucial features (e.g., house price or acreage); if the model follows this guidance, prediction quality will decrease. Or, suppose a user tries to add a word to a topic that does not appear in any of the documents; the model cannot add this word as it is out of vocabulary. In our study, refinements that did not work as expected were less popular (e.g., change word order and add word), whereas users preferred refinements that reflected their intent well (e.g., merge topics). Adherence is thus an important quality for developers of human-in-the-loop systems to consider. To account for this, when user input cannot be adhered to, transparent systems could either explain why or provide superficial adherence (i.e., treating word-level refinements as modifications of the model representation, which do not impact the underlying model).

Users might be willing to share control if they have a helpful partner. Importantly, our study also shows that users think about instability differently than the related concept of adherence. Instability was a lower priority consideration, and not all participants perceived it. For those who did, it was polarizing: some preferred “help” from the system, while others disliked it, particularly when model updates reverted prior changes (e.g., reintroducing previously removed words) or changed topics that users thought were already high quality. Therefore, our recommendation is to (1) better inform users to how models might update and clarifying why models might make other unexpected changes (i.e. faithfully modeling all underlying data); and (2) provide mechanisms for users to lock portions of the model which should not be updated and easily revert low quality, unstable updates. These recommendations should promote a healthier human-machine collaboration in which users and models can share control.

Different users, different needs. Users do not have a homogeneous process for interacting with models. As human-in-the-loop systems become more ubiquitous, designers should ensure that models and interfaces are robust to innate user variation. For example, while we did not explore this in our study, different levels of expertise, both with ML and the domain, could impact use: ML experts or those using the system on their own data are more likely to perceive when models update in unexpected ways, and while ML experts might be understanding of this, domain experts (without ML background), appear to become frustrated. Similarly, personality traits, such as confidence and locus of control, are likely to affect users’ desire to be in control, and increase their frustration if systems limit control.

Need for speed: latency and granularity. Machine learning pipelines typically focus on throughput as the metric of choice [17, 32]. This is indeed important for serving data-hungry models, but humans typically inspect high-level summaries rather than minutiae. Computational frameworks that can serve intermediate updates quickly would best address users’ complaints about “slowness”. Further, better management of latency expectations may have reduced frustration in our study; tutorials and initial introductions to ML tools should set expectations regarding latency, as well as other system attributes (e.g., instability and adherence).

Limitations. This study used a simple, and fairly short document organization task. Had participants been working with their own data, or working with the systems for longer periods of time, they might have been more invested in model quality, which in turn might have affected their perceptions and experience. Similarly, while our study was aimed at understanding how non-ML experts would best address users’ complaints about “slowness”. Further, better management of latency expectations may have reduced frustration in our study; tutorials and initial introductions to ML tools should set expectations regarding latency, as well as other system attributes (e.g., instability and adherence).

5 CONCLUSION

This paper explores users’ perceptions, experience, and behavior with easy-to-validate controls that vary in terms of control, particularly how well user input was adhered to and whether other changes occurred during model updates (instability), as well as how long updates took and model quality. We found that: (1) participants noticed—and often disliked—when their input was not adhered to, particularly for the easiest-to-validate refinements; (2) participants were polarized by instability, both in whether they noticed it and how they reacted to it: some participants liked it while others did not; (3) participants thought all the systems were slow but good: participants were satisfied with the final models they generated and thought they showed improvement over their starting points; (4) user experience did not differ between the systems: participants on average were confident in their input, trusted the models to update effectively, and thought the task was easy, but some participants were frustrated, particularly by slow updates.

6 ACKNOWLEDGMENTS

We thank the anonymous reviewers for their insightful and constructive comments. This work was supported by NSF Grant 11s-1409287 (umd and uw) and 11s-1409739 (byu). Any opinions, findings, conclusions, or recommendations expressed here are those of the authors and do not necessarily reflect the view of the sponsor.
REFERENCES


