

# TopicGPT: A Prompt-based Topic Modeling Framework

Chau Minh Pham<sup>1</sup> Alexander Hoyle<sup>2</sup> Simeng Sun<sup>1</sup> Philip Resnik<sup>2</sup> Mohit Iyyer<sup>1</sup>

<sup>1</sup>University of Massachusetts Amherst <sup>2</sup>University of Maryland  
{ctpham, simengsun, miyyer}@umass.edu,  
{hoyle, resnik}@umd.edu

## Abstract

Topic modeling is a well-established technique for exploring text corpora. Conventional topic models (e.g., LDA) represent topics as bags of words that often require “reading the tea leaves” to interpret; additionally, they offer users minimal control over the formatting and specificity of resulting topics. To tackle these issues, we introduce TopicGPT, a prompt-based framework that uses large language models (LLMs) to uncover latent topics in a text collection. TopicGPT produces topics that align better with human categorizations compared to competing methods: it achieves a harmonic mean purity of 0.74 against human-annotated Wikipedia topics compared to 0.64 for the strongest baseline. Its topics are also interpretable, dispensing with ambiguous bags of words in favor of topics with natural language labels and associated free-form descriptions. Moreover, the framework is highly adaptable, allowing users to specify constraints and modify topics without the need for model retraining. By streamlining access to high-quality and interpretable topics, TopicGPT represents a compelling, human-centered approach to topic modeling.<sup>1</sup>

## 1 Introduction

Topic modeling is a commonly used technique for discovering latent thematic structures in extensive collections of text documents. Traditional topic models such as latent Dirichlet allocation (Blei et al., 2003, LDA) represent documents as mixtures of topics, where each topic is a distribution over words. Topics are often represented with their most probable words, but this representation can contain incoherent or unrelated words that make topics difficult for users to interpret (Chang et al., 2009; Newman et al., 2010). Although some models enable users to interactively guide topics based on needs and domain knowledge (Hu et al., 2014; Nikolenko

et al., 2017), their usability is constrained by the bag-of-words topic format.

To address these limitations, we introduce TopicGPT (Figure 1), a human-centric approach to topic modeling that relies on prompting large language models to perform in-context topic generation (§3.1) and assignment (§3.2). First, we iteratively prompt an LLM to generate new topics given a sample of documents from an input dataset and a list of previously generated topics. The resulting set of topics can then be refined to eliminate redundant and infrequent topics. Finally, given a new document, an LLM assigns it to one or more of the generated topics, also providing a quotation from the document to support its assignment. These quotations make the method easily verifiable, addressing some validity concerns plaguing traditional topic models.

**TopicGPT produces higher-quality topics than competing approaches.** TopicGPT’s topics and assignments align significantly more closely with human-annotated ground truth topics than those from LDA, SeededLDA (Jagarlamudi et al., 2012), and BERTopic (Grootendorst, 2022) on two datasets: Wikipedia articles (Merity et al., 2018) and Congressional bills (as processed by Hoyle et al., 2022). We measure topical alignment using three external clustering metrics (harmonic mean purity, normalized mutual information, and adjusted Rand index) and find that TopicGPT improves substantially over baselines (e.g., absolute purity improves by 10 points over LDA on Wikipedia); furthermore, its topics are more semantically aligned with human-labeled topics (30.3% of TopicGPT’s topics are misaligned compared to 62.4% for LDA on Wikipedia). Further analyses demonstrate the robustness of TopicGPT’s topic quality across various prompt and data settings.

**TopicGPT produces more interpretable topics.** Topics generated by TopicGPT include natural lan-

<sup>1</sup>Code at <https://github.com/chtmp223/topicGPT>

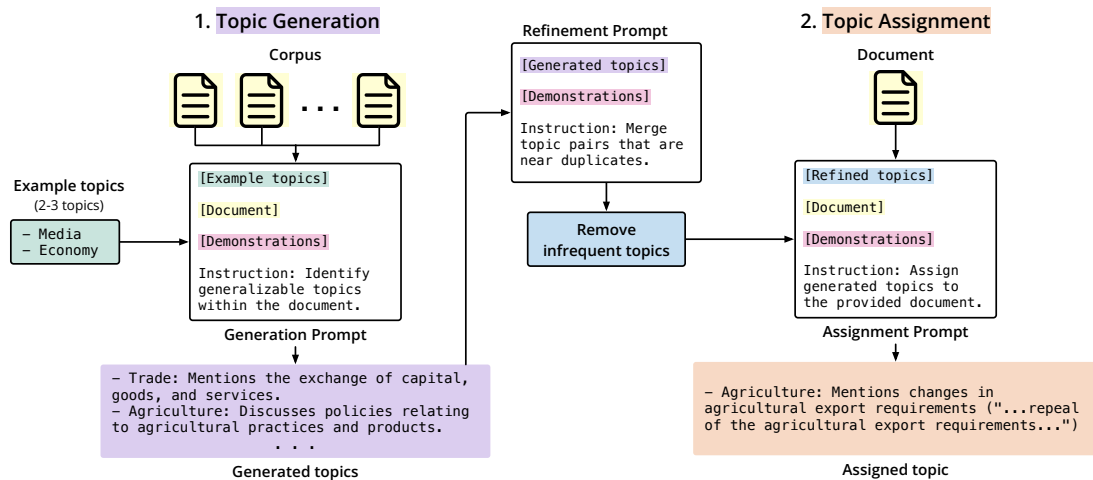


Figure 1: Overview of TopicGPT. **1) Topic Generation:** Given a corpus and some manually-curated example topics, TopicGPT identifies additional topics in each corpus document. The framework then refines the list by merging repeated topics and removing infrequent topics. **2) Topic Assignment:** Given the generated topics, TopicGPT assigns the most relevant topic to each document and provides a quote that supports this assignment.

guage labels and descriptions that make them immediately interpretable without needing a separate labeling step. Furthermore, the framework provides informative document-topic associations along with contextual quoted evidence. By creating intuitive topic structures and understandable document-topic assignments, TopicGPT aims to make the overall process interpretable.

**TopicGPT is customizable to fit user needs:** Topic models should suit their application settings (Doogan and Buntine, 2021). In addition to interpretability, TopicGPT offers users the ability to guide the generated topics to their specific needs. To steer the scope and formatting of the generated topics, users initially provide a small number of example topics, which need not be data-specific or representative of all possible topics in the corpus. After reviewing the initial results, users can manually edit or remove any topics to curate a coherent and goal-oriented list.

**Open-source LLMs are competent topic assigners but bad topic generators.** Most of our experiments implement topic generation with GPT-4 (OpenAI, 2023) and topic assignment with GPT-3.5-turbo, which costs around \$100 per dataset (Table 11). To remove the dependence on expensive APIs, we also experiment with the open-source Mistral-7B-Instruct model (Jiang et al., 2023), which assigns topics well but cannot competently follow instructions for generating topics. Thus, improving topic generation in open-source LLMs is

an important direction for future work.

## 2 Related Work

We designed TopicGPT for the use case of automated content analysis, which is one of the primary applications of topic models (Hoyle et al., 2022). In this methodology, practitioners first engage in open coding of individual documents to establish initial labels. These initial codes are then carefully examined, reconfigured, and organized into a cohesive coding system (Hsieh and Shannon, 2005; Kyngäs, 2020; Vears and Gillam, 2022).

**Topic modeling for content analysis:** Traditional approaches to topic modeling, such as Latent Dirichlet Allocation (Blei et al., 2003, LDA), are parameterized by topic-word and document-topic distributions that can reveal latent thematic structures in a corpus. However, these representations are not straightforward to interpret (Mei et al., 2007; Chang et al., 2009; Ramage et al., 2011), and subjective manual effort may result in issues of reliability and validity (Baden et al., 2021). Our work follows previous research that aims to produce more interpretable topics in natural language (Mei et al., 2007; Lau et al., 2011; Wan and Wang, 2016). Additionally, our design builds on seeded/anchored topic models (Andrzejewski and Zhu, 2009; Jagarlamudi et al., 2012; Gallagher et al., 2017; Meng et al., 2020), hierarchical topic models (Griffiths et al., 2003; Teh et al., 2006; Mimno et al., 2007; Paisley et al., 2014), as well as those that impose constraints on topics (Wallach et al.,

2009; Hu et al., 2014).

**LLM-based content analysis:** LLMs such as ChatGPT have enabled new prompting and embedding-based approaches to analyzing text. Researchers have used prompting techniques on these LLMs for related content analysis tasks, including text clustering (Viswanathan et al., 2023; Zhang et al., 2023; Hoyle et al., 2023), abstractive summarization (Liu and Healey, 2023), and deductive qualitative coding (Tai et al., 2023; Chew et al., 2023). Prior work has explored topic modeling with contextualized embeddings from pre-trained models (Sia et al., 2020; Thompson and Mimno, 2020; Bianchi et al., 2021; Grootendorst, 2022). Recent work has also used LLMs to label (Rijcken et al., 2023) and evaluate topics (Stammach et al., 2023) that are produced by an existing topic model.

**Comparison to GoalEx:** TopicGPT most closely resembles GoalEx (Wang et al., 2023), but is tailored specifically for topic modeling. GoalEx focuses on clustering and corpus *partitioning* but not overall topic set organization, which is a fundamental need for content analysis. Specifically, when refining topics, GoalEx only retains clusters such that each document is supported approximately once, which may remove topics that often appear together but have different meanings. Instead, our framework refines topics based on their semantics to retain coherent, informative topics. Additionally, where GoalEx assigns topics to documents individually, which scales poorly, TopicGPT simultaneously provides all prompt topics to each document for efficient scaling. While GoalEx was evaluated only on cluster recovery, TopicGPT is benchmarked on stability and alignment with ground-truth topics, demonstrating its usefulness as a content analysis tool beyond text clustering.

### 3 Methodology

TopicGPT consists of two main stages: topic generation (§3.1) and topic assignment (§3.2). Figure 1 provides an illustrative overview of our framework.

#### 3.1 Stage 1: Topic Generation

Broadly, we prompt an LLM to generate a set of topics given an input dataset, and then we further refine these topics to remove infrequently used ones and merge duplicates. The output of this step can optionally be fed into TopicGPT again to generate fine-grained subtopics (Appendix A).

**Generating new topics:** Given a document  $d$  from the corpus and a set of example topics  $S$ , the model is instructed to either assign  $d$  to an existing topic in  $S$  or generate a new topic that better describes  $d$  and add it to  $S$ . We define a *topic* to be a concise label paired with a broad one-sentence description, as in

**Trade:** Mentions the exchange of capital, goods, and services

where “Trade” serves as the topic label. Initially,  $S$  consists of a few human-written topics (our experiments use 2 example topics). These topics serve as “few-shot” demonstrations of the topic generation format. **Importantly, they do not need to be dataset-specific or representative of all possible topics in the corpus, as we show in Appendix D.** This iterative process encourages newly generated topics to be distinctive and match the specificity seen in other topics. Notably, instead of running topic generation over the entire corpus, which can be extremely costly, we apply the process to a carefully constructed sample from the dataset (§4.3).

**Refining generated topics:** Optionally, we can further refine the generated topics to ensure that the final topic list is coherent and non-redundant. We first use Sentence-Transformer embeddings (Reimers and Gurevych, 2019) to identify pairs of topics with cosine similarity  $\geq 0.5$ . We then prompt the LLM with five such topic pairs, instructing it to merge near-duplicate pairs where appropriate. To address any minor topics that may have been overlooked in the prior step, we eliminate topics with low frequency of occurrence. To do this, we keep track of how frequently each topic gets generated. If a topic occurs below a “removal” threshold frequency,<sup>2</sup> we consider that topic to be minor and remove it from the final list.

#### 3.2 Stage 2: Topic Assignment

In the assignment stage, we aim to establish a valid and interpretable association between the generated topics and the documents in our datasets. We provide the LLM with our generated topic list, 2-3 examples, and a document, the topic(s) of which we are interested in obtaining. We then instruct the model to assign one or more topics to the given document. The final output contains the assigned topic label, a document-specific topic description,

<sup>2</sup>We recommend trying different thresholds to make sure that important topics are not removed from the final list.

and a quote from the document that supports this assignment. The quoted text improves the verifiability of TopicGPT’s assignments, which has been a long-standing concern with traditional methods such as LDA. A sample topic assignment is

**Agriculture:** Mentions changes in agricultural export requirements (“...repeal of the agricultural export requirements...”)

where “Agriculture” is the assigned topic label, followed by the topic description and a quote from the document enclosed within parentheses.

**Self-correction:** To address topic assignments with incorrect formatting or low quality, we incorporate a self-correction step (Shinn et al., 2023; Sun et al., 2023). Specifically, we implement a parser to identify hallucinated or invalid assignments (e.g. “None”/“Error”). Next, we provide the LLM with the identified documents along with the error type and prompt the model to reassign a valid topic.

## 4 Experiments

Our goal is to assess whether TopicGPT’s outputs align with human-coded ground truth topics, as well as to test its robustness to various settings. Here, we describe our datasets, baseline methods, model configurations, and evaluation metrics.

### 4.1 Datasets

We use two English-language datasets for evaluation: `Wiki` and `Bills`. See Appendix H for statistics and discussion about data memorization.

**Wiki** (Merity et al., 2018) has 14,290 Wikipedia articles that meet a core set of editorial standards. This dataset comes with 15 high-level, 45 mid-level, and 279 low-level human-annotated labels.

**Bills** (Adler and Wilkerson 2018, compiled by Hoyle et al. 2022) contains 32,661 bill summaries from the 110 – 114<sup>th</sup> U.S. congresses, with 21 high-level and 114 low-level human-annotated labels.

### 4.2 Baselines

We consider three popular topic models that follow different paradigms: LDA, BERTopic, and SeededLDA. We control the number of topics  $k$  to be equal to the number of topics generated by TopicGPT for a fair comparison. We adopt Hoyle et al.

(2021)’s data preprocessing pipeline for LDA and SeededLDA. We do not preprocess for BERTopic.<sup>3</sup>

**LDA:** We use the MALLETT (McCallum, 2002) implementation of LDA with Gibbs sampling (Griffiths, 2002), which boasts strong alignment with human codes (Srivastava and Sutton, 2017; Hoyle et al., 2021) and high stability compared to neural topic models (Hoyle et al., 2022). We set  $|V| = 15,000$ ,  $\alpha = 1.0$ ,  $\beta = 0.1$ , and run LDA for 2,000 iterations with optimization at every 10 intervals.

**BERTopic:** BERTopic is a popular neural topic model that obtains topics by performing clustering on Sentence-Transformer embeddings of documents (Grootendorst, 2022). We maintain all default hyperparameters.

**SeededLDA:** SeededLDA (Jagarlamudi et al., 2012) incorporates seed topics to steer the resulting word-topic and document-topic distributions towards topics of interest to the users. We maintain all default hyperparameters

### 4.3 Sampling documents for TopicGPT

The number of documents used during the topic generation phase is a critical parameter of TopicGPT. Given enough time and money (if using closed-source LLM APIs), we could use the entire training corpus for topic generation. However, this is impractical given its high cost and also unnecessary as confirmed in practice below. We thus sample a document subset uniformly at random for topic generation, which results in 1,000 documents from `Bills` and 1,100 documents from `Wiki`.

**How many documents should we sample?** We recommend that users either choose a sample size that fits their budget or run topic generation incrementally and stop when no new topics are generated for some threshold (e.g., 200 documents). To assess the topic coverage of this approach, we examine the topics generated after reaching this threshold and check whether our refinement process removed any of them. In both datasets, the number of new topics that remain after refinement plateaus after this threshold - after around 600 documents have been generated.

### 4.4 TopicGPT implementation details

Our default TopicGPT setting uses OpenAI’s GPT-4 to generate topics and GPT-3.5-turbo to assign

<sup>3</sup><https://maartengr.github.io/BERTopic/faq.html#should-i-preprocess-the-data>

topics to documents.<sup>4</sup> To encourage deterministic outputs,<sup>5</sup> we set `max_tokens` to 300 and `temperature` and `top_p` to 0. We truncate longer documents to fit within the context window size of LLMs.<sup>6</sup> Since `Bills` and `Wiki` have one-to-one mappings between documents and labels, we modify the assigner prompt to assign only one topic per document, although we emphasize that the method does not require a single assignment. We set the removal frequency threshold to 10 and 5 for `Bills` and `Wiki`, respectively. We enable self-correction with a retry limit of 10 and end up resolving all topic hallucinations and formatting issues within this limit. For evaluation purposes, we sample 8,024 documents from `Wiki` and 15,242 documents from `Bills` that are not included in the topic generation sample.

## 4.5 Evaluation Setup

To assess the usefulness of TopicGPT as an automated content analysis tool, we evaluate the topic alignment and stability of the generated topics, following Hoyle et al. (2022).

### 4.5.1 Topical alignment

Since TopicGPT is not a probabilistic model like LDA, we compare it to baselines by assessing the alignment between predicted topic assignments and topic labels in the ground truth, following prior work (Chuang et al., 2013; Poursabzi-Sangdeh et al., 2016; Korenčić et al., 2021; Hoyle et al., 2022). For other methods, we assign each document to its most probable topic. Given a set of ground-truth classes and a set of predicted assignment clusters, we assessed alignment between the two sets using three external clustering metrics.

**Purity.** We use the harmonic mean of purity (Zhao, 2005) and inverse purity to match each ground-truth category with the cluster that has the highest combined precision and recall (Amigó et al., 2009). Purity yields a score of close to 0 for random assignments and nearly 1 for strongly consistent assignments.

**Adjusted Rand Index.** The Rand Index measures the pairwise agreement between two sets of

clusters. Adjusted Rand Index (ARI) further corrects for chance, yielding a score near 0 for random assignments and near 1 for consistent assignments (Hubert and Arabie, 1985; Vinh et al., 2009).

**Normalized Mutual Information.** Mutual Information (MI) measures the amount of shared information between two sets of clusters (Shannon, 1948). NMI normalizes MI to a value between 0 and 1, making MI less sensitive to a varying number of clusters (Strehl and Ghosh, 2002).

**Comparison of metrics:**  $P_1$ , ARI, and NMI provide complementary perspectives through set matching, counting pairs, and variation of information (Meilă, 2007).  $P_1$  emphasizes cluster purity, not so much the distribution of ground-truth labels. ARI, unlike NMI, is adjusted for chance, but it does not account for class imbalance, making it more sensitive to the number of predicted clusters.

### 4.5.2 Stability

We also assess the robustness of TopicGPT to changes in prompts and corpus samples for generation by measuring whether TopicGPT maintains high topical alignment in these modified settings.

**Out-of-domain prompts.** In TopicGPT, users can change the example topics and few-shot examples in the prompts to tailor the method to their dataset. We explore whether prompts written for one dataset can work on another by applying prompts for `Wiki` on the `Bills` dataset.

**Additional example topics.** We assess the impacts of additional example topics on TopicGPT’s performance in `Bills`. Our original prompt for topic generation has two example topics. We expand the prompt by adding three more topics.

**Shuffling sampled documents for topic generation.** We shuffle documents in the generation sample of `Bills` to understand the importance of the order in which LLM processes documents.

**Using a different sample for topic generation.** To evaluate TopicGPT’s robustness to data shift, we apply TopicGPT to a different generation sample from `Bills` and examine the results’ variation.

## 5 Results

Our results demonstrate TopicGPT’s strong topical alignment with ground truth and robustness to variations in prompts and data. Our human evaluation

<sup>4</sup>Results in the default settings were finalized in August 2023. Ablation results were finalized in October 2023.

<sup>5</sup>We note that OpenAI’s LLMs possess some degree of non-determinism even after fixing the decoding hyperparameters to perform greedy decoding.

<sup>6</sup>The context window size for GPT-4 is 8,192 tokens, whereas GPT-3.5-turbo has a context window of 4,096 tokens.

also show that the *semantic* content of TopicGPT’s generated topics (ignoring document assignment) is significantly more aligned with ground-truth topics on both datasets. Furthermore, we explore the use of an open-source model for topic generation.

### 5.1 TopicGPT is strongly aligned to ground truth labels

Our experiments (Table 1) show that TopicGPT identifies topics that are substantially more aligned with human-annotated labels than baselines, and that this improvement holds across all datasets, settings, and metrics. Of the three baselines, LDA is generally comparable to SeededLDA but superior to BERTopic for all metrics, suggesting that LDA remains a strong baseline. However, none of the baselines approaches the performance of TopicGPT: for example, TopicGPT achieves post-refinement harmonic purity scores  $P_1$  of 0.74 and 0.57 on `Wiki` and `Bills`, respectively, compared to 0.64 and 0.52 for LDA, 0.58 and 0.39 for BERTopic, and 0.62 and 0.52 for SeededLDA.

**Where does TopicGPT disagree with the ground truth?** To fully understand the disagreement between TopicGPT and human labels, we examine five assignments where the ground truth topics and TopicGPT’s assignments in the default setting do not match, and we find that each sampled document could reasonably be assigned multiple topics. Therefore, TopicGPT’s assignments can still be valid even if they differ from the ground truth labels. For example, the second document in Table 18 could fit either the “*Labor*” or “*Transportation Safety*” topics, though “*Labor*” is more prominent. If we allow TopicGPT to assign multiple topics per document, can it successfully retrieve all appropriate topics? To check, we re-run topic assignment using a prompt that allows the assignment of multiple topics per document. With this updated prompt, three out of the five `Wiki` examples are assigned to the ground truth as well as the originally assigned topic (Table 18). Thus, we recommend that practitioners use multi-label assignment prompts to extract as many relevant topics as possible.

### 5.2 TopicGPT is stable

Broadly speaking, TopicGPT produces comparable topical alignment with the ground truth across all modified experimental settings for the `Bills` data (lower portion of Table 1). The setting with additional example topics obtains the worst perfor-

mance, suggesting that simply adding more example topics is not always helpful. Too many example topics may overwhelm the model and lead to poorer coherence, as the model tries to fit diverse topics rather than consolidating around the most salient themes. We recommend keeping the example topic list small (2-3 high-quality topics) for best results, rather than lengthening the list arbitrarily.

**Consistency between multiple settings of TopicGPT:** To further evaluate the consistency of TopicGPT’s topic assignments, we computed alignment scores between the default setting and each modified setting. As a benchmark, we ran LDA 10 times with  $k=79$  topics and calculated the average internal alignment between each pair of runs. Table 2 shows TopicGPT’s assignments were highly stable across settings, with all metrics within a tight 0.05 range. TopicGPT demonstrated greater stability than LDA in terms of  $P_1$  and ARI, while achieving comparable NMI. Interestingly, TopicGPT produces slightly different outputs between two runs with identical settings, a possible result of adding randomness to the self-correction process (§3.2) as well as LLM API non-determinism.

### 5.3 TopicGPT topics are semantically close to ground truth

The metrics reported above do not capture whether the topics are semantically aligned with ground truth. To this end, we qualitatively compare LDA outputs and TopicGPT’s unrefined and refined topic generations and analyze the proportion of misaligned topics produced by each method.

**Manual topic matching process:** Three annotators (the first author and two external annotators)<sup>7</sup> went through the list of generated topics and assigned each topic to a ground truth class. If an exact match was not possible, the annotators labeled the generated topic as one of the three misaligned categories as below (Chuang et al., 2013):

1. Out-of-scope: topics that are too narrow/broad compared to the associated ground truth.
2. Missing topics: topics present in the ground truth but not in the generated outputs.
3. Repeated topics: topics that are duplicates of other topics.

<sup>7</sup>External annotators were recruited from our network and provided uncompensated annotations to support this research.

Dataset	Setting	TopicGPT			LDA			BERTopic			SeededLDA		
		$P_1$	ARI	NMI	$P_1$	ARI	NMI	$P_1$	ARI	NMI	$P_1$	ARI	NMI
Wiki	Default setting ( $k=31$ )	<b>0.73</b>	<b>0.58</b>	<b>0.71</b>	0.59	0.44	0.65	0.54	0.24	0.50	0.61	0.47	0.65
	Refined topics ( $k=22$ )	<b>0.74</b>	<b>0.60</b>	<b>0.70</b>	0.64	0.52	0.67	0.58	0.28	0.50	0.62	0.51	0.65
Bills	Default setting ( $k=79$ )	<b>0.57</b>	<b>0.42</b>	<b>0.52</b>	0.39	0.21	0.47	0.42	0.10	0.40	0.50	0.28	0.43
	Refined topics ( $k=24$ )	<b>0.57</b>	<b>0.40</b>	<b>0.49</b>	0.52	0.32	0.46	0.39	0.12	0.34	0.52	0.31	0.45
<i>TopicGPT stability ablations, baselines controlled to have the same number of topics (<math>k</math>).</i>													
Bills	Different generation sample ( $k=73$ )	<b>0.57</b>	<b>0.40</b>	<b>0.51</b>	0.41	0.23	0.47	0.38	0.08	0.38	0.40	0.21	0.44
	Out-of-domain prompts ( $k=147$ )	<b>0.55</b>	<b>0.39</b>	<b>0.51</b>	0.31	0.14	0.47	0.35	0.07	0.41	0.29	0.13	0.44
	Additional example topics ( $k=123$ )	<b>0.50</b>	<b>0.33</b>	<b>0.49</b>	0.33	0.15	0.46	0.36	0.07	0.40	0.33	0.15	0.44
	Shuffled generation sample ( $k=118$ )	<b>0.55</b>	<b>0.40</b>	<b>0.52</b>	0.33	0.16	0.47	0.36	0.08	0.40	0.34	0.18	0.44
	Assigning with Mistral ( $k=79$ )	<b>0.51</b>	<b>0.37</b>	0.46	0.39	0.21	<b>0.47</b>	0.42	0.10	0.40	0.50	0.28	0.43

Table 1: Topical alignment between ground-truth labels and predicted assignments. Overall, TopicGPT achieves the best performance across all settings and metrics compared to LDA, BERTopic, and SeededLDA. The number of topics used in each setting is specified as  $k$ . The largest values in each metric and setting are **bolded**.

Method	Setting	$P_1$	ARI	NMI
LDA	Default setting ( $k=79$ )	0.64	0.55	0.71
TopicGPT	Different generation sample ( $k=73$ )	0.67	0.61	0.69
	Out-of-domain prompts ( $k=147$ )	0.69	0.63	0.69
	Additional example topics ( $k=123$ )	0.69	0.59	0.70
	Shuffled generation sample ( $k=118$ )	0.70	0.63	0.70
	Running the pipeline twice ( $k=79$ )	0.95	0.92	0.92

Table 2: Stability of topic assignments of TopicGPT and LDA in the `Bills` dataset, as measured by the topical alignment between topic assignments of each modified setting against the default setting (unlike Table 1 which reports alignment against *ground-truth* assignments).

After completing the mapping, the percentage of misaligned topics was calculated for each annotator’s mappings (Table 3). Each person completed a total of 6 mappings, including LDA outputs, unrefined and refined topic list for both datasets. Mapping instructions and interface can be found in Appendix F.

**TopicGPT contains far fewer misaligned topics than LDA, especially after refinement.** Compared to LDA, both TopicGPT’s unrefined and refined topics are less likely to be misaligned overall (62.4% for LDA vs. 38.7% unrefined and 30.3% refined). Annotators also noted that TopicGPT’s outputs are much easier to work with compared to the ambiguous LDA outputs (see Table 4 for examples of LDA and TopicGPT’s outputs). We notice that refinement consistently reduces the number of out-of-scope and repeated topics. On the other hand, refinement does increase the number of missing topics by 1 in the Bills dataset; upon closer examination, this missing topic was “*Culture*”, which appears infrequently in the Bills corpus (only 23 documents out of 32,661 documents). This might be acceptable depending on the use case, and we

emphasize again that practitioners should try different refinement thresholds to avoid filtering out topics that are important to their research question. See Appendix B for an additional qualitative evaluation of TopicGPT in a different domain, compared with expert-derived qualitative categories.

#### 5.4 Implementing TopicGPT with open-source LLMs

We examine alternate LLMs for both topic assignment and topic generation and discover that while topic assignment can be feasibly performed with open-source LLMs, topic generation is too complex for all LLMs we tried other than GPT-4.

**Mistral-7B-Instruct for topic assignment:** We experiment with Mistral-7B-Instruct (Jiang et al., 2023) for topic assignment to assess its feasibility as a lower-cost alternative to GPT-3.5-turbo. Mistral’s topic assignments align reasonably well with human ground truth, though not to the same degree as GPT-3.5-turbo; the bottom row of Table 1 shows an absolute purity decrease of about  $\approx 6$  points. However, the resulting assignments still outperform baseline methods on  $P_1$  and ARI.

Dataset	Setting	Out-of-scope	Missing	Repeated	Total
Wiki	LDA ( $k=31$ )	46.3	4.3	11.9	62.4
	Unrefined ( $k=31$ )	38.7	<b>0.0</b>	1.1	39.8
	Refined ( $k=22$ )	<b>30.3</b>	<b>0.0</b>	<b>0.0</b>	<b>30.3</b>
Bills	LDA ( $k=79$ )	56.1	2.1	22.0	80.2
	Unrefined ( $k=79$ )	65.0	<b>1.3</b>	3.8	70.1
	Refined ( $k=24$ )	<b>27.8</b>	4.2	<b>0.0</b>	<b>31.9</b>

Table 3: Comparison of misaligned topic proportions between LDA and TopicGPT outputs. Values are averaged over three annotations and rounded to one decimal place. Across both datasets, TopicGPT (with refinement) achieves the lowest proportion of misaligned topics. The best (lowest) misalignment proportion for each dataset is **bolded**.

Data	Document	Ground truth	TopicGPT assignment	LDA assignment
Wiki	Grant Park Music Festival = The Grant Park Music Festival ( formerly Grant Park Concerts ) is an annual ten-week classical music concert series held in Chicago, Illinois, USA. It features the Grant Park Symphony Orchestra and Grant Park Chorus along with featured guest performers and conductors. The Festival has earned non-profit organization status. It claims to be the nation’s only free, outdoor classical music series. The Grant Park Music Festival has been a Chicago tradition since 1931 when Chicago Mayor Anton Cermak suggested free concerts to lift the spirits of...	Music	<b>Music &amp; Performing Arts:</b> Discuss creation, production, and performance of music, as well as related arts and cultural aspects.	<b>City infrastructure:</b> city, building, area, new, park
Bills	Perkins Fund for Equity and Excellence. This bill amends the Carl D. Perkins Career and Technical Education Act of 2006 to replace the existing Tech Prep program with a new competitive grant program to support career and technical education. Under the program, local educational agencies and their partners may apply for grant funding to support: career and technical education programs that are aligned with postsecondary education programs, dual or concurrent enrollment programs and early college programs, certain evidence-based strategies and delivery models related to career and technical education, teacher and leader experiential ...	Education	<b>Education:</b> Mentions policies and programs related to higher education and student loans.	<b>Programs and grants:</b> program, grants, grant, programs, state

Table 4: Example topic assignments from TopicGPT and LDA (showing top 5 topic words) on two documents. While TopicGPT’s topics closely align with the ground truth, LDA’s topics are influenced by frequently occurring words, causing it to overlook the overarching theme of the document. TopicGPT’s topic labels and descriptions are both automatically generated, while LDA produces a bag of words that needs to be **manually labeled**.

**Mistral-7B-Instruct for topic generation:** We further test both Mistral and GPT-3.5-turbo as topic generation models, finding that both models struggle to follow formatting instructions for topic generation. In total, Mistral and GPT-3.5-turbo produced 1,418 and 151 topics, respectively. The large number of topics made it impossible to include all of them in a single topic assignment prompt. Additionally, most of the generated topics are overly specific with a low frequency of occurrence, meaning they would likely be removed during refinement. We emphasize that the instructions for topic generation are complex with many criteria (Table 14). As such, we recommend sticking with GPT-4 or models with similar capabilities to generate topics.

## 6 Future Work

TopicGPT is designed for inductive content analysis, assuming the user has some prior familiarity with the dataset’s content. However, the framework can also be applied to data exploration scenarios where the user is unfamiliar with the dataset and

employs topic models to gain insights. Future work could explore the potential of zero-shot prompting (without providing example topics or documents) with TopicGPT for such exploratory use cases.

Appendix A presents a hierarchical extension of this framework. Evaluating the performance of this extension against established hierarchical topic models through rigorous comparative studies will be a valuable next step.

## 7 Conclusion

We introduce a prompt-based framework, TopicGPT, specifically designed for topic modeling. TopicGPT addresses traditional topic models’ interpretability and adaptability limitations by generating high-quality and descriptive topics. Our results demonstrate that TopicGPT outperforms baseline topic models in terms of topic alignment with ground truth labels while also showing robustness across different prompts and data subsets. We release our pipeline so that interested researchers and practitioners can try our framework.



## Limitations

### Transparency concerns of closed-source models.

TopicGPT achieves optimal performance using GPT-4 for topic generation and GPT-3.5-turbo for assignment, both of which are closed-source LLMs. Unfortunately, we have limited transparency into their pre-training and instruction tuning datasets, as well as their architectural details. Future work can explore using a stronger open-source model for topic generation or fine-tuning an LLM for topic assignment. Our reliance on closed-source LLMs for topic generation reflects the current imbalance between the instruction-following abilities of closed and open-source models rather than a permanent limitation. We hope this will eventually be addressed by the rapid advances in open-source LLMs.

**Cost concerns of closed-source models.** TopicGPT’s use of closed-source models also incurs costs for each run (see Table 11 for our expenses). In addition to exploring open-source alternatives to these models, users can take advantage of the framework’s modular design and tailor it to their use cases. For instance, if users are solely interested in obtaining a list of topics associated with a dataset, they can skip topic refinement and assignment steps or reduce the size of the corpus subset used for topic generation, thus saving running time and money.

**Dealing with context limits.** Another limitation of our current approach is the need to truncate documents to fit TopicGPT’s context length limit. By only providing partial documents, we lose potentially valuable context and risk misrepresenting the contents of full documents. While truncation was necessary in these initial experiments, we recognize it is not an ideal solution. Future work can explore strategies to represent full documents within length limits, such as incrementally feeding in chunks of the document, sampling representative chunks, or providing a summarized version of the document. Additionally, future work can explore using long-context LLMs like GPT-4-turbo (128k tokens),<sup>8</sup> Claude (200k tokens),<sup>9</sup> or LLaMA-2-7B-32K (32K tokens)<sup>10</sup> in the framework.

<sup>8</sup><https://platform.openai.com/docs/models/gpt-4-and-gpt-4-turbo>

<sup>9</sup><https://www.anthropic.com/news/claude-3-family>

<sup>10</sup><https://huggingface.co/togethercomputer/LLaMA-2-7B-32K>

**Multilinguality.** We have not yet evaluated TopicGPT on non-English datasets. However, OpenAI’s LLMs are pre-trained and instruction-tuned primarily on English language data, and the instruction-following capabilities of LLMs in non-English languages are thus notably degraded (Huang et al., 2023; Li et al., 2023). We hope that future advances in multilingual LLMs will make TopicGPT more broadly accessible.

## Ethical Considerations

The risks posed by our framework are no greater than those inherent in the large language models that support it (Weidinger et al., 2021). Our human evaluation received approval from an institutional review board. All annotators (US-based) gave their informed consent and participated voluntarily, without compensation, to support our research.

## Acknowledgements

We are grateful to Dzung Pham for the helpful discussion on evaluation metrics, Maria Antoniak, Brendan O’Connor, and Laure Thompson for their insights into topic model usage and evaluation, Chau-Anh and Upwork annotators for their assistance with the qualitative evaluation, and the UMass NLP community for helpful feedback throughout the project. This project was partially supported by awards IIS-2202506 and IIS-2046248 from the National Science Foundation (NSF).

## References

- E Scott Adler and John Wilkerson. 2018. Congressional bills project: 1995-2018.
- Enrique Amigó, Julio Gonzalo, Javier Artiles, and Felisa Verdejo. 2009. A comparison of extrinsic clustering evaluation metrics based on formal constraints. *Information retrieval*, 12:461–486.
- David Andrzejewski and Xiaojin Zhu. 2009. *Latent Dirichlet Allocation with Topic-in-Set Knowledge*. In *Proceedings of the NAACL HLT 2009 Workshop on Semi-supervised Learning for Natural Language Processing*, pages 43–48, Boulder, Colorado. Association for Computational Linguistics.
- Christian Baden, Christian Pipal, Martijn Schoonvelde, and Mariken A.C.G. van der Velden. 2021. Three gaps in computational text analysis methods for social sciences: A research agenda. *Communication Methods and Measures*, 16:1 – 18.
- Federico Bianchi, Silvia Terragni, and Dirk Hovy. 2021. Pre-training is a hot topic: Contextualized document embeddings improve topic coherence.

- David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *Journal of machine Learning research*, 3(Jan):993–1022.
- Jonathan Chang, Sean Gerrish, Chong Wang, Jordan Boyd-Graber, and David Blei. 2009. Reading tea leaves: How humans interpret topic models. *Advances in neural information processing systems*, 22.
- Robert Chew, John Bollenbacher, Michael Wenger, Jessica Speer, and Annice Kim. 2023. [Llm-assisted content analysis: Using large language models to support deductive coding.](#)
- Jason Chuang, Sonal Gupta, Christopher Manning, and Jeffrey Heer. 2013. [Topic model diagnostics: Assessing domain relevance via topical alignment.](#) In *Proceedings of the 30th International Conference on Machine Learning*, volume 28 of *Proceedings of Machine Learning Research*, pages 612–620, Atlanta, Georgia, USA. PMLR.
- Caitlin Doogan and Wray Buntine. 2021. [Topic model or topic twaddle? re-evaluating semantic interpretability measures.](#) In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3824–3848, Online. Association for Computational Linguistics.
- Ryan J. Gallagher, Kyle Reing, David Kale, and Greg Ver Steeg. 2017. [Anchored Correlation Explanation: Topic Modeling with Minimal Domain Knowledge.](#) *Transactions of the Association for Computational Linguistics*, 5:529–542.
- Thomas Griffiths, Michael Jordan, Joshua Tenenbaum, and David Blei. 2003. Hierarchical topic models and the nested chinese restaurant process. *Advances in neural information processing systems*, 16.
- Tom Griffiths. 2002. Gibbs sampling in the generative model of latent dirichlet allocation. *Stanford University*, 518(11):1–3.
- Maarten Grootendorst. 2022. [Bertopic: Neural topic modeling with a class-based tf-idf procedure.](#)
- Alexander Hoyle, Pranav Goel, Andrew Hian-Cheong, Denis Peskov, Jordan Boyd-Graber, and Philip Resnik. 2021. [Is Automated Topic Model Evaluation Broken? The Incoherence of Coherence.](#) In *Advances in Neural Information Processing Systems*, volume 34, pages 2018–2033. Curran Associates, Inc.
- Alexander Hoyle, Rupak Sarkar, Pranav Goel, and Philip Resnik. 2023. [Natural language decompositions of implicit content enable better text representations.](#)
- Alexander Miserlis Hoyle, Pranav Goel, Rupak Sarkar, and Philip Resnik. 2022. [Are neural topic models broken?](#) In *Findings of the Association for Computational Linguistics: EMNLP 2022*, pages 5321–5344, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.
- Hsiu-Fang Hsieh and Sarah E Shannon. 2005. Three approaches to qualitative content analysis. *Qualitative health research*, 15(9):1277–1288.
- Yuening Hu, Jordan Boyd-Graber, Brianna Satinoff, and Alison Smith. 2014. Interactive topic modeling. *Machine learning*, 95:423–469.
- Haoyang Huang, Tianyi Tang, Dongdong Zhang, Wayne Xin Zhao, Ting Song, Yan Xia, and Furu Wei. 2023. Not all languages are created equal in llms: Improving multilingual capability by cross-lingual-thought prompting. In *Findings of Empirical Methods in Natural Language Processing*.
- Lawrence Hubert and Phipps Arabie. 1985. Comparing partitions. *Journal of classification*, 2:193–218.
- Jagadeesh Jagarlamudi, Hal Daumé III, and Raghavendra Udupa. 2012. [Incorporating Lexical Priors into Topic Models.](#) In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 204–213, Avignon, France. Association for Computational Linguistics.
- Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Léo Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. [Mistral 7b.](#)
- Damir Korenčić, Strahil Ristov, Jelena Repar, and Jan Šnajder. 2021. [A Topic Coverage Approach to Evaluation of Topic Models.](#) *IEEE Access*, 9:123280–123312. ArXiv:2012.06274 [cs].
- Helvi Kyngäs. 2020. Inductive content analysis. *The application of content analysis in nursing science research*, pages 13–21.
- Jey Han Lau, Karl Grieser, David Newman, and Timothy Baldwin. 2011. Automatic labelling of topic models. In *Proceedings of the 49th annual meeting of the association for computational linguistics: human language technologies*, pages 1536–1545.
- Haonan Li, Fajri Koto, Minghao Wu, Alham Fikri Aji, and Timothy Baldwin. 2023. [Bactrian-x : A multilingual replicable instruction-following model with low-rank adaptation.](#)
- Marsha M Linehan, Judith L Goodstein, Stevan L Nielsen, and John A Chiles. 1983. Reasons for staying alive when you are thinking of killing yourself: the reasons for living inventory. *Journal of consulting and clinical psychology*, 51(2):276.
- Sengjie Liu and Christopher G Healey. 2023. Abstractive summarization of large document collections using gpt. *arXiv preprint arXiv:2310.05690*.
- Andrew Kachites McCallum. 2002. Mallet: A machine learning for language toolkit. [Http://www.cs.umass.edu/mccallum/mallet](http://www.cs.umass.edu/mccallum/mallet).

- Qiaozhu Mei, Xuehua Shen, and ChengXiang Zhai. 2007. [Automatic labeling of multinomial topic models](#). In *Knowledge Discovery and Data Mining*.
- Marina Meilă. 2007. Comparing clusterings—an information based distance. *Journal of multivariate analysis*, 98(5):873–895.
- Yu Meng, Jiaxin Huang, Guangyuan Wang, Zihan Wang, Chao Zhang, Yu Zhang, and Jiawei Han. 2020. [Discriminative Topic Mining via Category-Name Guided Text Embedding](#). In *Proceedings of The Web Conference 2020*, pages 2121–2132. ArXiv:1908.07162 [cs].
- Stephen Merity, Nitish Shirish Keskar, and Richard Socher. 2018. [Regularizing and optimizing LSTM language models](#). In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.
- David Mimno, Wei Li, and Andrew McCallum. 2007. Mixtures of hierarchical topics with pachinko allocation. In *Proceedings of the 24th international conference on Machine learning*, pages 633–640.
- David Newman, Jey Han Lau, Karl Grieser, and Timothy Baldwin. 2010. Automatic evaluation of topic coherence. In *Human language technologies: The 2010 annual conference of the North American chapter of the association for computational linguistics*, pages 100–108.
- Sergey I Nikolenko, Sergei Koltcov, and Olessia Koltsova. 2017. Topic modelling for qualitative studies. *Journal of Information Science*, 43(1):88–102.
- OpenAI. 2023. [Gpt-4 technical report](#).
- John Paisley, Chong Wang, David M Blei, and Michael I Jordan. 2014. Nested hierarchical dirichlet processes. *IEEE transactions on pattern analysis and machine intelligence*, 37(2):256–270.
- Forough Poursabzi-Sangdeh, Jordan Boyd-Graber, Leah Findlater, and Kevin Seppi. 2016. Alto: Active learning with topic overviews for speeding label induction and document labeling. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1158–1169.
- Daniel Ramage, Christopher D. Manning, and Susan Dumais. 2011. [Partially labeled topic models for interpretable text mining](#). In *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 457–465, San Diego California USA. ACM.
- William M Rand. 1971. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical association*, 66(336):846–850.
- Nils Reimers and Iryna Gurevych. 2019. [Sentence-BERT: Sentence embeddings using Siamese BERT-networks](#). In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.
- Philip Resnik, Pranav Goel, Alexander Hoyle, Rupak Sarkar, Josh Hagedorn, Maeve Gearing, and Carol Bruce. 2022. A step-by-step protocol for curation of topic models by subject matter experts. In *New Directions in Analyzing Text as Data*.
- Philip Resnik, Bolei Ma, Alexander Hoyle, Pranav Goel, Rupak Sarkar, Maeve Gearing, Anna-Carolina Haensch, and Frauke Kreuter. 2024a. A topic-oriented protocol for content analysis of text. Manuscript in preparation.
- Philip Resnik, Katherine Musacchio Schafer, Rebecca Resnik, Josh Hagedorn, and Jonathan Singer. 2024b. Reasons to live instead of dying by suicide: New insights from a computer assisted content analysis. Manuscript in preparation.
- Emil Rijcken, Floortje Scheepers, Kalliopi Zervanou, Marco Spruit, Pablo Mosteiro, and Uzay Kaymak. 2023. Towards Interpreting Topic Models with ChatGPT: The 20th World Congress of the International Fuzzy Systems Association.
- Claude Elwood Shannon. 1948. A mathematical theory of communication. *The Bell system technical journal*, 27(3):379–423.
- Noah Shinn, Federico Cassano, Beck Labash, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. 2023. [Reflexion: Language agents with verbal reinforcement learning](#).
- Suzanna Sia, Ayush Dalmia, and Sabrina J. Mielke. 2020. [Tired of topic models? clusters of pretrained word embeddings make for fast and good topics too!](#) In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1728–1736, Online. Association for Computational Linguistics.
- Akash Srivastava and Charles Sutton. 2017. Autoencoding variational inference for topic models. *arXiv preprint arXiv:1703.01488*.
- Dominik Stambach, Vilém Zouhar, Alexander Hoyle, Mrinmaya Sachan, and Elliott Ash. 2023. [Revisiting automated topic model evaluation with large language models](#).
- Alexander Strehl and Joydeep Ghosh. 2002. Cluster ensembles—a knowledge reuse framework for combining multiple partitions. *Journal of machine learning research*, 3(Dec):583–617.

- Simeng Sun, Yang Liu, Shuohang Wang, Chenguang Zhu, and Mohit Iyyer. 2023. [Pearl: Prompting large language models to plan and execute actions over long documents](#).
- Robert H Tai, Lillian R Bentley, Xin Xia, Jason M Sitt, Sarah C Fankhauser, Ana M Chicas-Mosier, and Barnas G Monteith. 2023. Use of large language models to aid analysis of textual data. *bioRxiv*, pages 2023–07.
- Yee Whye Teh, Michael I Jordan, Matthew J Beal, and David M Blei. 2006. [Hierarchical Dirichlet Processes](#). *Journal of the American Statistical Association*, 101(476):1566–1581.
- Laure Thompson and David Mimno. 2020. [Topic Modeling with Contextualized Word Representation Clusters](#). ArXiv:2010.12626 [cs].
- Danya F Vears and Lynn Gillam. 2022. Inductive content analysis: A guide for beginning qualitative researchers. *Focus on Health Professional Education: A Multi-disciplinary Journal*, 23(1):111–127.
- Nguyen Xuan Vinh, Julien Epps, and James Bailey. 2009. Information theoretic measures for clusterings comparison: is a correction for chance necessary? In *Proceedings of the 26th annual international conference on machine learning*, pages 1073–1080.
- Vijay Viswanathan, Kiril Gashteovski, Carolin Lawrence, Tongshuang Wu, and Graham Neubig. 2023. [Large language models enable few-shot clustering](#).
- Hanna Wallach, David Mimno, and Andrew McCallum. 2009. Rethinking lda: Why priors matter. *Advances in neural information processing systems*, 22.
- Xiaojun Wan and Tianming Wang. 2016. [Automatic labeling of topic models using text summaries](#). In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2297–2305, Berlin, Germany. Association for Computational Linguistics.
- Zihan Wang, Jingbo Shang, and Ruiqi Zhong. 2023. [Goal-driven explainable clustering via language descriptions](#).
- Laura Weidinger, John Mellor, Maribeth Rauh, Conor Griffin, Jonathan Uesato, Po-Sen Huang, Myra Cheng, Mia Glaese, Borja Balle, Atoosa Kasirzadeh, et al. 2021. Ethical and social risks of harm from language models. *arXiv preprint arXiv:2112.04359*.
- Yuwei Zhang, Zihan Wang, and Jingbo Shang. 2023. [ClusterLLM: Large Language Models as a Guide for Text Clustering](#). ArXiv:2305.14871 [cs].
- Ying Zhao. 2005. *Criterion Functions for Document Clustering*. Ph.D. thesis, University of Minnesota, USA. AAI3180039.

## A Hierarchical Implementation

TopicGPT can be further extended to hierarchical topical modeling, enabling users to explore topics at various levels of granularity. Specifically, we treat the generated topics that remain after the refinement stage as *top-level topics* and prompt the LLM for more specific subtopics at subsequent levels. We then provide the model with a *topic branch* that contains a top-level topic  $t$ , a list of example subtopics  $S'$ , and the documents  $d_t$  associated with the top-level topic  $t$ . With these inputs, we instruct the LLM to generate subtopics  $t'$  that capture common themes among the provided documents. The model must also return specific documents supporting each subtopic to ensure subtopics are grounded in the documents rather than hallucinated. If the documents cannot fit into a single prompt, we divide them into different prompts and include subtopics generated by earlier prompts in subsequent ones.

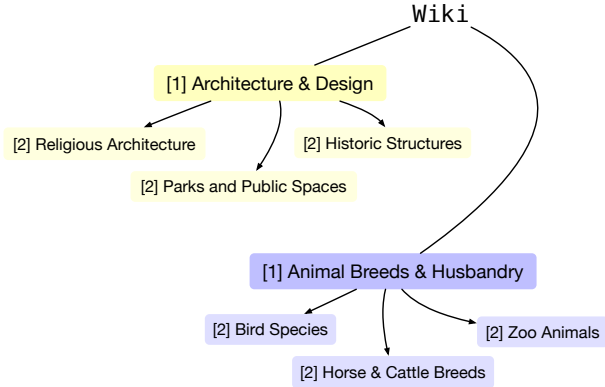


Figure 2: Example topic hierarchy for `Wiki`, with "Architecture & Design" and "Animal Breeds & Husbandry" as the top-level topics generated by TopicGPT. This hierarchical topic structure, in which the upper-level topics are broad enough to encompass more detailed subtopics, allows users to explore topics at different levels of specificity.

We apply TopicGPT hierarchically to our datasets and find that the generated subtopics are informative and well-grounded in the documents associated with their parent topics. These subtopics successfully capture more narrow and nuanced themes within the broader parent topic, allowing for richer exploration and analysis compared to a flat topic list. Figure 2 provides a closer look into a portion of the generated topic hierarchy for the `Wiki` dataset.

## B Comparison of TopicGPT topics with human-curated qualitative categories in a different domain

As an additional form of evaluation, we conducted a case study directly comparing TopicGPT with categories in a different domain (Resnik et al., 2024b). These categories were obtained using TOPCAT, a qualitative content analysis protocol involving structured human curation of automatically obtained topical categories (Resnik et al., 2022, 2024a).<sup>11</sup> The dataset comprised more than 16K responses to a question on Reddit asking formerly suicidal Redditors what had gotten them through their dark times.<sup>12</sup> A 30-topic LDA model, created using preprocessing essentially identical to the preprocessing described in Section 4.2, served as input to curation by experts in the study of and/or prevention of suicide, resulting in the 24 categories in Table 5.<sup>13</sup> TopicGPT produced the categories in Table 6.

Qualitatively, TopicGPT has succeeded in capturing many aspects of the expert-derived category analysis. Denoting expert categories as  $E_i$  and TopicGPT categories as  $T_j$ , we see:

- Direct correspondences. These include  $T_1$  with  $E_8$ ,  $T_6$  with  $E_{21}$ ,  $T_{11}$  with  $E_4$ , and arguably  $T_8$  with  $E_{10}$  and  $T_{14}$  with  $E_7$ .
- Cross-cutting but reasonable categories. Categories  $T_4$ ,  $T_{12}$ , and  $T_{13}$  all bear a clearly visible relationship to  $E_1$ – $E_5$ . All of these represent variations on a general theme of concern about others who are left behind, which is the essence of the experts’ first metacategory, although the experts and TopicGPT carve up the space in different ways.
- Higher level themes. This includes  $T_5$ , which (ignoring the oddly specific example in the description) is a higher-level category in relation to  $E_{16}$ ,  $E_{17}$ ,  $E_{18}$ ,  $E_{20}$ , and arguably  $E_{19}$ . It may also be an alternative characterization of cross-cutting categories.

There are also some categories that were not among the expert-derived categories. These all appear to capture ideas that are valid in terms of characterizing themes present for many responses in the document collection but are not directly responsive to the question:  $T_2$ ,  $T_3$ , and  $T_{10}$  all appear to characterize background content rather than categories of responses *per se* to the question that was asked, and  $T_9$  characterizes non-answers. This highlights the fact that qualitative content analysis is never driven just by the data; it is also driven by the research questions behind the data analysis (Resnik et al., 2022, 2024a,b). We would argue that these TopicGPT categories can nonetheless be useful, both because they potentially provide insight into themes about background information that could potentially be correlated with reasons individuals chose not to die by suicide, and because having well-defined "irrelevant" categories, particularly  $T_9$  ("junk" responses), may contribute to useful ways to filter and/or further break down the collection of responses.

We further implement our human evaluation protocol (Section 5.3) on the dataset with two subject matter experts recruited via Upwork. Table 7 shows the average percentage of missing, out-of-scope, and repeated topics generated by TopicGPT and SeededLDA relative to the expert-curated ground truth. Unlike the protocol in Section 5.3, we compare with SeededLDA instead of LDA because LDA is used to craft the ground-truth expert categories, which may introduce bias. While TopicGPT and SeededLDA have similar proportions of out-of-scope and repeated topics, TopicGPT has fewer missing topics, indicating broader coverage. Both experts note that TopicGPT is superior to SeededLDA in terms of usability and interpretability, particularly because TopicGPT outputs follow a format that is coherent and “more closely related to standardized codes in thematic analysis”.

<sup>11</sup>Selecting the model size (number of topics) is part of the protocol.

<sup>12</sup>[https://www.reddit.com/r/AskReddit/comments/j0z4lp/formerly\\_suicidal\\_redditors\\_whats\\_something\\_that/](https://www.reddit.com/r/AskReddit/comments/j0z4lp/formerly_suicidal_redditors_whats_something_that/). This work received an Exempt determination from our Institutional Review Board.

<sup>13</sup>Resnik et al. (2024b) situates these categories in relationship to theories of suicide, contrasts them with Linehan’s (1983) widely used Reasons for Living inventory, and discusses potential clinical implications.

---

**Expert "reasons to live" categories**

- E1 Not wanting to hurt specific family members
  - E2 Financial burdens for survivors or if attempt fails
  - E3 Not wanting to traumatize loved ones, especially finding and having to deal with the body
  - E4 Pets, especially cats and dogs – not wanting to hurt them, worry about them, etc.
  - E5 Not wanting to hurt specific family members, particularly children
  
  - E6 Holding on for one more day and focusing on the next thing, routine, sleep on it and see if/how things get better in the morning
  - E7 Quotes, inspiration, philosophy, wonder – things that resonate/connect
  - E8 Sense of connection – feeling seen, heard, understood, cared about by other people
  - E9 Professional diagnosis and treatment
  - E10 Hope and connection with and/or fear, from spiritual or religious beliefs
  - E11 Recognition that suicide is "a permanent solution to a temporary problem" and life will change
  - E12 Hope – looking past the bad and feeling like life can get better
  - E13 Insight/realization about not wanting to die or finding meaning in surviving an attempt
  - E14 Professional diagnosis and treatment
  - E15 Insights from reading or writing, including writing a suicide note
  
  - E16 Looking forward to next episodes of TV, new games, experiencing new media
  - E17 Music – wanting to continue hearing it, experiencing music as meaningful, specific artists
  - E18 Food and sensory pleasures
  - E19 Substance use and abuse
  - E20 Spending time on alternative media
  - E21 Turning point, transition, new connection
  - E22 Finding a new distraction or focus
  
  - E23 Fear of surviving an attempt and being worse off
  - E24 Spite – proving people wrong
- 

Table 5: Categories obtained using a topic-oriented protocol for content analysis of text, applied to more than 16,000 responses from formerly suicidal individuals about what got them through their dark times (Resnik et al., 2024b). Two subject matter experts independently followed a structured series of steps to curate a 30-topic LDA model, followed by a consensus process yielding 24 relevant categories. Further expert analysis grouped these into four themes or meta-categories.

---

**TopicGPT "reasons to live" categories**

- T1 Emotionally supportive close relationships: The respondent cites emotional support from family, a significant other, or friend.
  - T2 Emotional distress: The respondent expresses feelings of emotional distress, including feelings of being unappreciated, belittled, and harassed.
  - T3 Social alienation: The respondent feels alienated or ostracized due to differing opinions or beliefs.
  - T4 Fear of causing harm to others: The respondent is concerned about the negative impact their death would have on their loved ones.
  - T5 Therapeutic hobbies: The respondent finds solace and joy in sharing pictures of corgis, which helps them cope with their emotional distress.
  - T6 Personal growth and goal setting: The respondent finds purpose and direction in setting personal goals and working towards them, such as pursuing a career in veterinary medicine.
  - T7 Seeking professional help: The respondent mentions seeking help from a professional, such as a therapist or counselor.
  - T8 Fear of the unknown: The respondent expresses uncertainty or fear about what happens after death.
  - T9 Junk response: The respondent does not reply to the prompt or does not take the response seriously.
  - T10 Loss of a loved one: The respondent is dealing with the death of someone close to them.
  - T11 Responsibility towards pets: The respondent mentions the need to care for a pet, which includes providing food and affection.
  - T12 Death would be a burden: The respondent believes that their death would have caused distress to others.
  - T13 Responsibility towards others: The respondent feels obligated to keep a promise made to someone else.
  - T14 Appreciation of life: The respondent expresses a deep appreciation for life and its beauty.
- 

Table 6: Categories obtained automatically using TopicGPT for the same responses from formerly suicidal individuals about what got them through their dark times, for comparison with the expert categories in Table 5.

	TopicGPT	SeededLDA
Out-of-scope	25%	25%
Missing	<b>47.82%</b>	56.52%
Repeated	21.43%	21.43%

Table 7: Comparison of misaligned topic proportions between SeededLDA ( $k = 14$ ) and unrefined TopicGPT outputs on Reason to Live dataset. Values are averaged over three annotations and rounded to two decimal places. The best (lower) misalignment proportion is **bolded**.



## C Probabilistic Justification for Data Sampling

We plot the number of topics generated over documents processed in the `Bills` and `Wiki` corpus (Figure 3). The graph confirms that after an initial "topic drought" period where few new topics are discovered, the number of initially generated topics continues increasing as more documents are processed. However, the final refined topics plateau over time, reaching a saturation point regardless of additional documents seen.

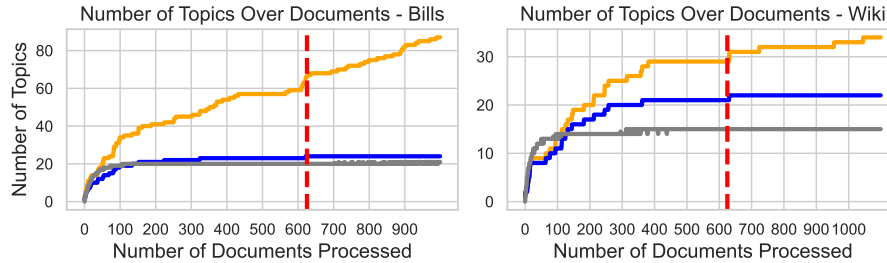


Figure 3: The number of topics generated over documents processed in the `Bills` and `Wiki` corpus. The grey line indicates the number of expected topics, simulated using the empirical distribution of ground-truth topics for the datasets. For both datasets, we see a similar pattern - after a "topic drought" period marked by the dashed red line, the number of initially generated topics (orange line) keeps increasing. However, the final refined topics (blue line) and expected number of topics (grey line) plateau, despite more documents being processed.

We present a probabilistic approach to sampling data for topic generation. The user sets a lower bound on the number of documents that should be assigned to the least-prevalent topic,  $n_d$ , which induces an upper bound on the number of topics,  $K_u = \lfloor N/n_d \rfloor$ . We can then model the sample of size  $n_s$  as a draw from a uniform<sup>14</sup>  $\mathbf{c} \sim \text{Multi}(n_s, K_u)$ , where the expected number of zeros is then  $\frac{(K_u - 1)^{n_s}}{K_u^{n_s} - 1}$ . A straightforward simulation can be used to find the  $n_s$  that minimizes  $p^* = |P(\min_k c_k = 0) - \varepsilon|$ , where  $\varepsilon$  is a user-defined acceptable minimum probability of failing to find the least-prevalent topic. Taking the `Wiki` dataset, if we set  $n_d = 140$  (1% of the corpus) then  $n_s = 1,100$  leads to  $p^* \approx 0.005$ .

<sup>14</sup>If it were non-uniform, then we would have a topic with fewer than  $n_d$  documents.

## D Example Topics

In all levels of topic generation, we include a list of example topics along with corresponding example documents. These example topics can be curated to inform the formatting and specificity of resulting topics. In this section, we offer suggestions on crafting example topics and provide corresponding illustrations of their impacts on the resulting topics.

1. The format of the example topics should resemble the desirable output format, which contains a topic level and a concise topic label.

*Example 1: Using long example topic labels*

Example topics:

[1] Trade Policies

[1] Agricultural Policies

Generated topics (using GPT-4):

[1] Social Security Policies

[1] Food Safety Policies

*Example 2: Using short example topic labels.* Example topics:

[1] Trade

[1] Agriculture

Generated topics (using GPT-4):

[1] Social Security

[1] Food Safety

2. The example topics should match the degree of granularity you would expect in a given hierarchy level

*Example 1: Using abstract example topics for level 2.*

Example topics:

[1] Music & Performing Arts

- [2] Music

- [2] Performing Arts

Generated topics (using GPT-4):

[1] Engineering & Technology

- [2] Civil & Transportation Engineering

- [2] Electrical Engineering

*Example 2: Using detailed example topic for level 2.*

Example topics:

[1] Music & Performing Arts

- [2] Albums

- [2] Songs

Generated topics (using GPT-4):

[1] Engineering & Technology

- [2] Road & Highway Systems

- [2] Microwave Technology

## E Varying number of topics $k$ for LDA baseline

In the original experiments, we compare TopicGPT, which does not require  $k$  to be specified, to baselines (LDA, BERTopic, SeededLDA) that do require  $k$  to be specified. We choose to set  $k$  to the number of topics returned by TopicGPT to ensure a fair comparison with respect to our metrics. We further justify our decision with experiments that demonstrate TopicGPT’s superior topic alignment regardless of  $k$  values.

We compare TopicGPT in the default and refined settings to LDA across different values of  $k$  (15, 25, 50, 75, 100). As shown in Table 8, TopicGPT (with and without refinement) outperforms LDA on external clustering metrics in all settings. This experiment shows that regardless of  $k$ , TopicGPT has better alignment with ground-truth topics compared to LDA.

Data	Setting	P1	ARI	NMI
Bills (k=21)	TopicGPT (default setting, k=79)	<b>0.57</b>	<b>0.42</b>	<b>0.52</b>
	TopicGPT (refined topics, k=24)	0.57	0.40	0.49
	LDA (k=15)	0.51	0.32	0.44
	LDA (k=25)	0.53	0.32	0.47
	LDA (k=50)	0.48	0.27	0.48
	LDA (k=75)	0.40	0.22	0.47
	LDA (k=100)	0.35	0.18	0.47
Wiki (k=15)	TopicGPT (default setting, k=31)	0.73	0.58	<b>0.71</b>
	TopicGPT (refined topics, k=22)	<b>0.74</b>	<b>0.60</b>	0.70
	LDA (k=15)	0.67	0.58	0.69
	LDA (k=25)	0.61	0.48	0.66
	LDA (k=50)	0.51	0.36	0.63
	LDA (k=75)	0.43	0.26	0.61
	LDA (k=100)	0.39	0.22	0.60

Table 8: Performance of TopicGPT and LDA of various  $k$  values. The best performance in each data and metric setting is highlighted.

## F Human annotations on topical alignment between TopicGPT and ground truth of Bills dataset

In this section, we provide additional details on our human evaluation (Section 5.3). Table 9 shows the proportion of misaligned topics determined by each annotator. Table 10 and Figure 4 show the mapping and interface of the human annotation task.

Dataset	Setting	Out-of-scope			Missing			Repeated			Total		
		P1	P2	P3	P1	P2	P3	P1	P2	P3	P1	P2	P3
Wiki	LDA ( $k = 31$ )	74.2	19.4	45.2	3.2	6.5	3.2	9.7	6.5	19.4	87.1	32.4	67.8
	Unrefined ( $k = 31$ )	67.7	25.8	<b>22.7</b>	<b>0.0</b>	<b>0.0</b>	<b>0.0</b>	3.2	<b>0.0</b>	<b>0.0</b>	70.9	25.8	22.7
	Refined ( $k = 22$ )	<b>59.1</b>	<b>9.1</b>	<b>22.7</b>	<b>0.0</b>	<b>0.0</b>	<b>0.0</b>	<b>0.0</b>	<b>0.0</b>	<b>0.0</b>	<b>59.1</b>	<b>9.1</b>	<b>22.7</b>
Bills	LDA ( $k = 79$ )	64.6	50.6	53.2	2.5	<b>2.5</b>	1.3	16.5	22.8	26.6	83.6	75.9	81.0
	Unrefined ( $k = 79$ )	72.2	62.0	60.8	<b>1.3</b>	<b>2.5</b>	<b>0.0</b>	6.3	2.5	2.5	79.8	67.1	63.3
	Refined ( $k = 24$ )	<b>45.8</b>	<b>4.2</b>	<b>33.3</b>	4.2	4.2	4.2	<b>0.0</b>	<b>0.0</b>	<b>0.0</b>	<b>50.0</b>	<b>8.3</b>	<b>37.5</b>

Table 9: Proportion of misaligned topics in LDA’s outputs as well as TopicGPT’s unrefined and refined outputs (rounded to one decimal place). The lowest values in each column are bolded.

### Mapping Instruction for Human Annotators

**Introduction** We ask you to compare categories created by two topic models with those created by humans for the same dataset. This comparison will help us understand whether these methods effectively cover relevant topics and accurately represent them. Additionally, we request that you provide a short discussion (1-2 paragraphs) for each method on how helpful and understandable the categories are for each of the two methods.

#### Terminologies

- Out-of-scope categories: These are categories that are irrelevant compared to the true categories. For example, “Albums” and “Songs” would be too narrow compared to a true category of “Music”.
- Repeated categories: These are categories that are unnecessary repeats or very similar to others. For example, “year month time ago day life” and “time hours month cycle year” are near duplicates.
- Missing categories: These are essential categories that appear in the true categories but are absent in the categories generated by the coding schemes.

#### Instructions

0. The sheet name (at the bottom) indicates the method used to get these categories. Method A and C represents categories in natural language, with category labels and descriptions. Method B represents categories in terms of keywords, and you need to decide what the category is about. For example, the set of keywords “year month time ago day life” could be interpreted as referring to a “Time” category.

1. For each category in the first column, choose 1, 2, or 3 corresponding true categories. For example, suppose Method A produced a category “Outdoor activities and sports”, and suppose two of the true categories were “Outdoor sports” and “Enjoying being outside” (not a real example). In this case you might choose to include both true categories.

If the category is too specific or too general compared to the associated true categories, check the box in the “Out-of-scope” column. If there are categories that are unnecessary repeats or very similar to others, only one instance of the category should be left alone and all the other instances should be marked as “Repeated”. The number of “Missing” will be automatically calculated (you will not be required to do anything about this).

2. Provide a brief comment here (1-2 paragraphs) regarding the usability and interpretability of generated categories in corpus exploration and analysis. Here are some example questions to consider. Please note the specific method when making your comments.

Table 10: Instructions on the mapping process, provided in the form of a Google Doc for human annotators.

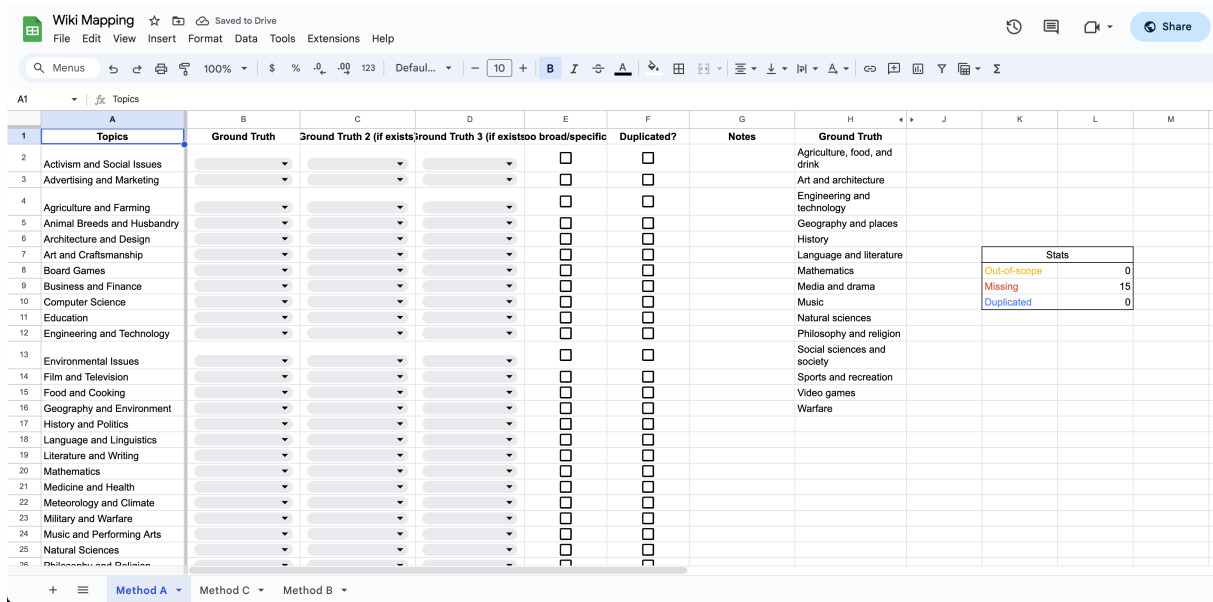


Figure 4: Mapping interface for human annotators. After reading the instructions in 10, the annotators perform mapping between generated and ground-truth topics.

## G Topical alignment metrics

Here, we provide more details on the external clustering metrics used to measure topical alignment.

**Purity.** Purity measures the degree to which predicted clusters contain data points predominantly from a single ground-truth class (Zhao, 2005). Purity is computed by assigning each predicted cluster to the ground-truth class that occurs most frequently within it, and then counting the number of correctly assigned documents and dividing by the number of clustered items  $N$ ,

$$\text{Purity}(\Omega, \mathbf{C}) = \sum_k \frac{|\omega_k|}{N} \max_j \text{Precision}(\omega_k, c_j) \quad (1)$$

where the precision of a cluster  $\omega_k$  for a given ground-truth class  $\mathbf{C}_i$  is calculated as:

$$\text{Precision}(\omega_k, c_j) = \frac{|\omega_k \cap c_j|}{|\omega_k|} \quad (2)$$

While high purity indicates low intra-cluster noise, it does not reward grouping items from the same ground-truth class. Purity reaches its maximum value of 1 when each cluster contains just one item. Inverse Purity, defined as  $\text{Purity}^{-1} = \text{Purity}(\mathbf{C}, \Omega)$ , addresses this by rewarding clustering data points from the same class together. However, Inverse Purity fails to penalize mixing items from different ground-truth categories. Amigó et al. (2009) show that the harmonic mean of Purity and Inverse Purity balances these two objectives:

$$P_1 = \sum_k \frac{|c_k|}{N} \max_j F(c_j, \omega_k)$$

where

$$F(c_k, \omega_j) = \frac{2 * \text{Precision}(c_k, \omega_j) * \text{Recall}(c_k, \omega_j)}{\text{Precision}(c_k, \omega_j) + \text{Recall}(c_k, \omega_j)} \quad (3)$$

and

$$\text{Recall}(\mathbf{C}, \Omega) = \text{Precision}(\Omega, \mathbf{C}) \quad (4)$$

**Adjusted Rand Index.** The Rand Index measures the pairwise agreement between two sets of clusterings (Rand, 1971):

$$RI(\Omega, \mathbf{C}) = \frac{TP + TN}{TP + FP + FN + TN} \quad (5)$$

where  $TP$  is the number of true positives,  $TN$  is the number of true negatives,  $FP$  is the number of false positives, and  $FN$  is the number of false negatives. Adjusted Rand Index (ARI) further corrects for chance by comparing it to the expected value

$$ARI(\Omega, \mathbf{C}) = \frac{RI - \mathbb{E}[RI]}{\max(RI) - \mathbb{E}[RI]} \quad (6)$$

where  $\mathbb{E}[RI]$  stands for expected rand index (Hubert and Arabie, 1985; Vinh et al., 2009). ARI yields a score close to 0 for random cluster assignments and near 1 for strongly consistent assignments.

**Normalized Mutual Information.** Mutual Information (MI) measures the amount of shared information between two sets of clusterings (Shannon, 1948). Normalized Mutual Information (NMI) normalizes the MI score to a value between 0 and 1, making MI less sensitive to a varying number of clusters (Strehl and Ghosh, 2002):

$$NMI(\Omega, \mathbf{C}) = \frac{I(\Omega, \mathbf{C})}{[H(\Omega) + H(\mathbf{C})]/2} \quad (7)$$

where  $I(\Omega, \mathbf{C})$  is the mutual information between  $\Omega$  and  $\mathbf{C}$ ,  $H(\Omega)$  and  $H(\mathbf{C})$  are the entropy of  $\Omega$  and  $\mathbf{C}$ .

## H Experiment datasets

Table 12 shows high-level statistics for `Wiki` and `Bills` datasets. Table 11 details the cost of running TopicGPT on each dataset.

Stages	Bills	Wiki
Topic generation	\$30	\$90
Topic refinement	\$10	\$5
Assignment + self-correction	\$48	\$60
Total	\$88	\$155

Table 11: Estimated cost (in US dollars) of running TopicGPT on the `Bills` and `Wiki` datasets. Though smaller in size, the `Wiki` dataset incurred a much higher cost to run TopicGPT than the `Bills` dataset due to its longer document length - on average 3,412 tokens/document compared to just 261 tokens/document in `Bills`.

	Bills	Wiki
# docs	32,661	14,290
avg. # tokens / doc	261	3,412
# test examples	15,242	8,024
# topic generation docs	1,000	1,100

Table 12: Dataset statistics of `Bills` and `Wiki`. The average number of tokens per document was calculated using `tiktoken` BPE tokenizer.<sup>15</sup>

**Potential effects of data memorization on TopicGPT’s performance** `Bills` is not likely to appear in LLMs’ training data because the label text is not directly associated with the corresponding documents. Specifically, the zipped data on [www.congressionalbills.org/download.html](http://www.congressionalbills.org/download.html) that connects bills to labels contains numerical codes for the labels and URLs pointing to the bills.

On the other hand, since `Wiki` text-label mapping might have appeared in the pre-training corpus of LLMs, TopicGPT’s performance on this dataset reflects a best-case scenario for our algorithm (Thompson and Mimno, 2020). However, even in this potential case, TopicGPT does not directly reproduce any ground truth topics from `Wiki` unless explicitly provided as example topics. Furthermore, the performance on `Wiki` is not perfect, indicating that simply memorizing ground truth topics cannot fully explain the results. Our novel prompting template, along with providing human-curated topic examples, means the model outputs are more guided by these few-shot demonstrations rather than resembling verbatim memorization.

<sup>15</sup><https://github.com/openai/tiktoken>

## I Additional seeded/anchored topic models

We include here the description and results of two additional seeded/anchored topics models: CorEx (Gallagher et al., 2017) and BERTopic-guided (Grootendorst, 2022).

**Seed words:** For all seeded topic models (CorEx, BERTopic-guided, and SeededLDA), we use the following seed words:

- **Bills:** ('trade', 'practices', 'products') and ('agriculture', 'capital', 'goods', 'services').
- **Wiki:** ('natural', 'sciences', 'biology', 'chemistry', 'physics') and ('engineering', 'system', 'infrastructure').

**CorEx:** CorEx (Gallagher et al., 2017) dispenses with the generative assumptions of LDA and instead uses total correlation to learn topics. Anchored CorEx incorporates user-provided seed words to create topics that balance between the original corpus and the information from the seed topics.

**BERTopic-guided:** BERTopic-guided is the seeded variant of BERTopic (Grootendorst, 2022). BERTopic assigns user-provided seed words to some documents based on their embeddings' cosine similarity. These assignments are then combined with UMAP to create a self-supervised approach, nudging topic creation to the seed topics.

Table 13 shows the topic alignment results for CorEx, SeededLDA, BERTopic-guided. Overall, CorEx and SeededLDA are slightly better than BERTopic-guided. Between CorEx and SeededLDA, CorEx is better when the number of topics is high ( $k > 100$  in `bills`). However, when  $k < 100$ , SeededLDA outperforms CorEx. SeededLDA is comparable to LDA, but they all did not come close to TopicGPT.

Dataset	Setting	TopicGPT			BERTopic-guided			CorEx			SeededLDA		
		$P_1$	ARI	NMI	$P_1$	ARI	NMI	$P_1$	ARI	NMI	$P_1$	ARI	NMI
Wiki	Default setting ( $k=31$ )	<b>0.73</b>	<b>0.58</b>	<b>0.71</b>	0.26	0.03	0.21	0.52	0.3	0.46	0.61	0.47	0.65
	Refined topics ( $k=22$ )	<b>0.74</b>	<b>0.60</b>	<b>0.70</b>	0.26	0.03	0.19	0.51	0.27	0.46	0.62	0.51	0.65
Bills	Default setting ( $k=79$ )	<b>0.57</b>	<b>0.42</b>	<b>0.52</b>	0.30	0.05	0.35	0.38	0.19	0.31	0.50	0.28	0.43
	Refined topics ( $k=24$ )	<b>0.57</b>	<b>0.40</b>	<b>0.49</b>	0.26	0.04	0.22	0.37	0.17	0.29	0.52	0.31	0.45
<i>TopicGPT stability ablations, baselines controlled to have the same number of topics (<math>k</math>).</i>													
Bills	Different generation sample ( $k=73$ )	<b>0.57</b>	<b>0.40</b>	<b>0.51</b>	0.32	0.07	0.35	0.35	0.16	0.3	0.40	0.21	0.44
	Out-of-domain prompts ( $k=147$ )	<b>0.55</b>	<b>0.39</b>	<b>0.51</b>	0.27	0.05	0.37	0.38	0.18	0.33	0.29	0.13	0.44
	Additional example topics ( $k=123$ )	<b>0.50</b>	<b>0.33</b>	<b>0.49</b>	0.29	0.05	0.37	0.35	0.17	0.30	0.33	0.15	0.44
	Shuffled generation sample ( $k=118$ )	<b>0.55</b>	<b>0.40</b>	<b>0.52</b>	0.30	0.05	0.37	0.42	0.24	0.34	0.34	0.18	0.44
	Assigning with Mistral ( $k=79$ )	<b>0.51</b>	<b>0.37</b>	<b>0.47</b>	0.30	0.05	0.35	0.38	0.19	0.31	0.50	0.28	0.43

Table 13: Topical alignment between ground-truth labels and predicted assignments. Overall, TopicGPT achieves the best performance across all settings and metrics compared to SeededLDA, CorEx, and BERTopic-guided. The number of topics used in each setting is specified as  $k$ . The largest values in each metric and setting are **bolded**.



## J Prompts

See Table 14, 15, 16, 17 for the prompt templates that we used in our framework.

---

### Prompt template for generating first-level/flat topics

---

You will receive a document and a set of top-level topics from a topic hierarchy. Your task is to identify generalizable topics within the document that can act as top-level topics in the hierarchy. If any relevant topics are missing from the provided set, please add them. Otherwise, output the existing top-level topics as identified in the document.

[Top-level topics]

{Example topics (containing "[1] Trade" in this example)}

[Examples]

Example 1: Adding "[1] Agriculture"

Document:

Saving Essential American Sailors Act or SEAS Act - Amends the Moving Ahead for Progress in the 21st Century Act (MAP-21) to repeal the Act's repeal of the agricultural export requirements that: (1) 25 of the gross tonnage of certain agricultural commodities or their products exported each fiscal year be transported on U.S. commercial vessels, and (2) the Secretary of Transportation (DOT) finance any increased ocean freight charges incurred in the transportation of such items. Revives and reinstates those repealed requirements to read as if they were never repealed.

Your response:

[1] Agriculture: Mentions policies relating to agricultural practices and products.

Example 2: Duplicate "[1] Trade", returning the existing topic

Document:

Amends the Harmonized Tariff Schedule of the United States to suspend temporarily the duty on mixtures containing Fluopyram.

Your response:

[1] Trade: Mentions the exchange of capital, goods, and services.

[Instructions]

Step 1: Determine topics mentioned in the document.

- The topic labels must be as GENERALIZABLE as possible. They must not be document-specific.
- The topics must reflect a SINGLE topic instead of a combination of topics.
- The new topics must have a level number, a short general label, and a topic description.
- The topics must be broad enough to accommodate future subtopics.

Step 2: Perform ONE of the following operations:

1. If there are already duplicates or relevant topics in the hierarchy, output those topics and stop here.
2. If the document contains no topic, return "None".
3. Otherwise, add your topic as a top-level topic. Stop here and output the added topic(s). DO NOT add any additional levels.

[Document]

{Document}

Please ONLY return the relevant or modified topics at the top level in the hierarchy.

[Your response]

---

Table 14: Prompt template for generating broad, high-level topics that can either serve as a flat list of standalone topics or as the first tier of a hierarchical topic taxonomy. Users should use this template in the typical, non-hierarchical use case. The designation of 'first-level' ensures these topics are sufficiently expansive to cover the topic distribution of the entire dataset. In practice, users need to modify the components highlighted in red (example topic list and document) as well as tailor the examples to their specific dataset.

---

**Prompt template for generating second-level subtopics**

---

You will receive a branch from a topic hierarchy along with some documents assigned to the top-level topic of that branch. Your task is to identify generalizable second-level topics that can act as subtopics to the top-level topic in the provided branch. Add your topic(s) if they are missing from the provided branch. Otherwise, return the existing relevant or duplicate topics.

[Example] (Return "[2] Exports" (new) and "[2] Tariff" (existing) as the subtopics of "[1] Trade" (provided).)

Topic branch:

- [1] Trade
  - [2] Tariff
  - [2] Foreign Investments

Document 1:

Export Promotion Act of 2012 - Amends the Export Enhancement Act of 1988 to revise the duties of the Trade Promotion Coordinating Committee (TPCC). Requires the TPCC to: (1) make a recommendation for the annual unified federal trade promotion budget to the President; and (2) review the proposed fiscal year budget of each federal agency with responsibility for export promotion or export financing activities before it is submitted to the Office of Management and Budget (OMB) and the President, when (as required by current law) assessing the appropriate levels and allocation of resources among such agencies in support of such activities.

Document 2:

Amends the Harmonized Tariff Schedule of the United States to suspend temporarily the duty on mixtures containing Fluopyram.

Document 3:

Securing Exports Through Coordination and Technology Act - Amends the Foreign Relations Authorization Act, Fiscal Year 2003. Requires carriers obliged to file Shipper's Export Declarations to file them through AES (either directly or through intermediaries) before items are exported from any U.S. port, unless the Secretary of Commerce grants an exception.

Your response:

- [1] Trade
  - [2] Exports (Document: 1, 3): Mentions export policies on goods.
  - [2] Tariff (Document: 2): Mentions tax policies on imports or exports of goods.

[Instructions]

Step 1: Determine PRIMARY and GENERALIZABLE topics mentioned in the documents.

- The topics must be generalizable among the provided documents.
- Each topic must not be too specific so that it can accommodate future subtopics.
- Each topic must reflect a SINGLE topic instead of a combination of topics.
- Each top-level topic must have a level number and a short label. Second-level topics should also include the original documents associated with these topics (separated by commas) as well as a short description of the topic.
- The number of topics proposed cannot exceed the number of documents provided.

Step 2: Perform ONE of the following operations:

1. If the provided top-level topic is specific enough, DO NOT add any subtopics. Return the provided top-level topic.
2. If your topic is duplicate or relevant to the provided topics, DO NOT add any subtopics. Return the existing relevant topic.
3. If your topic is relevant to and more specific than the provided top-level topic, add your topic as a second-level topic. DO NOT add to the first or third level of the hierarchy.

[Topic branch]

{Topic}

[Documents]

{Documents}

DO NOT add first- or third-level topics.

[Your response]

---

Table 15: Prompt template for generating second-level subtopics in a topic hierarchy. In practice, users need to modify the components highlighted in red (generated topic branch and associated documents) as well as tailor the examples to their specific dataset. This prompt can be further accommodate subtopic generation for lower-level by changing the topic level in the examples and instructions.

---

**Prompt template for refining (merging) topics**

---

You will receive a list of topics that belong to the same level of a topic hierarchy. Your task is to merge topics that are paraphrases or near duplicates of one another. Return "None" if no modification is needed.

[Examples]

Example 1: Merging topics ("[1] Employer Taxes" and "[1] Employment Tax Reporting" into "[1] Employment Taxes")

Topic List:

- [1] Employer Taxes: Mentions taxation policy for employer
- [1] Employment Tax Reporting: Mentions reporting requirements for employer
- [1] Immigration: Mentions policies and laws on the immigration process
- [1] Voting: Mentions rules and regulation for the voting process

Your response:

[1] Employment Taxes: Mentions taxation report and requirement for employer ([1] Employer Taxes, [1] Employment Tax Reporting)

Example 2: Merging topics ([2] Digital Literacy and [2] Telecommunications into [2] Technology)

- [2] Mathematics: Discuss mathematical concepts, figures and breakthroughs.
- [2] Digital Literacy: Discuss the ability to use technology to find, evaluate, create, and communicate information.
- [2] Telecommunications: Mentions policies and regulations related to the telecommunications industry, including wireless service providers and consumer rights.

Your response

[2] Technology: Discuss technology and its impact on society. ([2] Digital Literacy, [2] Telecommunications)

[Rules]

- Each line represents a topic, with a level indicator and a topic label.
- Perform the following operations as many times as needed:
- Merge relevant topics into a single topic.
- Do nothing and return "None" if no modification is needed.
- When merging, the output format should contain a level indicator, the updated label and description, followed by the original topics.

[Topic List]

{Topics}

Output the modification or "None" where appropriate. Do not output anything else.

[Your response]

---

Table 16: Prompt template for refining/merging similar topics in a topic hierarchy. In practice, users need to modify the components highlighted in red (example topic list and document) to contain the similar topic pairs.

---

### Prompt template for assigning topics

---

You will receive a document and a topic hierarchy. Assign the document to the most relevant topics the hierarchy. Then, output the topic labels, assignment reasoning and supporting quotes from the document. DO NOT make up new topics or quotes.

Here is the topic hierarchy:

{tree}

[Examples]

Example 1: Assign "[1] Agriculture" to the document

Document:

Saving Essential American Sailors Act or SEAS Act - Amends the Moving Ahead for Progress in the 21st Century Act (MAP-21) to repeal the Act's repeal of the agricultural export requirements that: (1) 25% of the gross tonnage of certain agricultural commodities or their products exported each fiscal year be transported on U.S. commercial vessels, and (2) the Secretary of Transportation (DOT) finance any increased ocean freight charges incurred in the transportation of such items.

Your response:

[1] Agriculture: Mentions changes in agricultural export requirements ("...repeal of the agricultural export requirements that...")

Example 2: Assign "[2] Tariff" to the document

Document:

Amends the Harmonized Tariff Schedule of the United States to suspend temporarily the duty on mixtures containing Fluopyram.

Your response:

[1] Trade

[2] Tariff: Mentions adjusting the taxation on mixtures containing Fluopyram ("...suspend temporarily the duty on mixtures containing Fluopyram.")

[Instructions]

1. Topic labels must be present in the provided topic hierarchy. You MUST NOT make up new topics.
2. The quote must be taken from the document. You MUST NOT make up quotes.
3. If the assigned topic is not on the top level, you must also output the path from the top-level topic to the assigned topic.

[Document]

{Document}

Double check that your assignment exists in the hierarchy!

[Your response]

---

Table 17: Prompt template for assigning topics to a given document in the corpus. In practice, users need to modify the components highlighted in red as well as tailor the examples to the specific dataset (can be reused from prompts in previous stages). Users can also modify the prompt to strictly assign to one topic.

Dataset	Document	Ground-truth label	Initial label	Reassigned labels
Bills	Securing Health for Ocean Resources and Environment Act or the SHORE Act - Requires the Under Secretary for Oceans and Atmosphere to: (1) review the National Oceanic and Atmospheric Administration's (NOAA) capacity to respond to oil spills; (2) be responsible for developing and maintaining oil spill trajectory modeling capabilities...	Technology	Environment	Technology; Environment
Bills	Driver Fatigue Prevention Act. This bill amends the Fair Labor Standards Act of 1938 to apply its maximum hours requirements to over-the-road bus drivers.	Labor	Transportation Safety	Labor; Transportation Safety
Bills	Defense Travel Simplification Act of 2007 - Requires the Secretary of Defense to: (1) redesignate the Defense Travel System as the Defense Travel Accounting and Voucher Processing System; and (2) establish an intra-agency task force to recommend measures to streamline and simplify the commercial travel system...	Domestic Commerce	State and Local Government	Technology; Military and Veterans Affairs
Bills	Amends the Internal Revenue Code to allow until June 30, 2010: (1) a first-time homebuyer tax credit for all purchasers of a principal residence (not just first-time homebuyers); and (2) a refundable tax credit, up to \$3,000, for the costs of refinancing a principal residence.	Domestic Commerce	Housing	Housing
Bills	Amends the Internal Revenue Code to extend through 2014 the equalization of the exclusion from gross income for employer-provided mass transit and parking benefits.	Labor	Transportation Safety	Taxation; Transportation Safety
Wiki	Colonel Cyrus Kurtz Holliday (April 3, 1826 – March 29, 1900) was one of the founders of the township of Topeka, Kansas, in the mid 19th century; and was Adjutant General of Kansas during the American Civil War. The title Colonel, however, was honorary. He was the first president of the Atchison, Topeka and Santa Fe Railway, as well as one of the railroad's directors for nearly 40 years, up to 1900...	Engineering and Technology	History and Politics	History and Politics; Engineering and Technology
Wiki	Jack Banham Coggins (July 10, 1911 – January 30, 2006) was an artist, author, and illustrator. He is known in the United States for his oil paintings, which focused predominantly on marine subjects. He is also known for his books on space travel, which were both authored and illustrated by Coggins. Besides his own works, Coggins also provided illustrations for advertisements, magazine covers and articles....	Language and literature	Art and Craftmanship	Art and Craftmanship; Literature and Writing
Wiki	HMS Belfast is a museum ship, originally a Royal Navy light cruiser, permanently moored in London on the River Thames and operated by the Imperial War Museum. Construction of Belfast, the first Royal Navy ship to be named after the capital city of Northern Ireland, and one of ten Town-class cruisers, began in December 1936...	Warfare	History and Politics	Military and Warfare
Wiki	The Grand Street Bridge was a double-leaf deck-girder bascule bridge in Bridgeport, Connecticut, United States, that spanned the Pequonnock River and connected Grand Street and Artic Street. It was one of three movable bridges planned by the City of Bridgeport in 1916 at the request of the War Department during World War I....	Art and architecture	Engineering and Technology	Engineering and Technology
Wiki	Burger King Specialty Sandwiches = The Burger King Specialty Sandwiches are a line of sandwiches developed by the international fast-food restaurant chain Burger King in 1978 and introduced in 1979 as part of a new product line designed to expand Burger King's menu with more sophisticated, adult-oriented fare beyond hamburgers...	Agriculture, food, and drink	Business and Finance	Food and Cooking; Business and Finance; Advertising and Marketing

Table 18: Error analysis on five examples from each of Bills and Wiki datasets. Documents are truncated for ease of viewing.