Textonomy: A TnT-LLM-Based Approach for Interpretable Topic Modeling at Scale

Anonymous ACL submission

Abstract

Automating text content analysis via topic modeling with Large Language Models (LLM) faces a trilemma: a trade-off between interpretability, scalability, and the accessibility of open-source models. This paper argues for a task-oriented view of topic modeling and introduces Textonomy[1], an implementation of the two-stage TnT-LLM framework, as a practical solution. Textonomy first uses an LLM to iteratively generate a data-driven taxonomy from a small sample of document summaries. It then trains a lightweight classifier on LLM-generated pseudo-labels for efficient, large-scale inference. We conduct a rigorous evaluation against traditional (LDA), neural (BERTopic), and pure-LLM (TopicGPT) topic models on two distinct datasets: WikiText-103 and a corpus of US Congressional bills. To address reproducibility, we benchmark Textonomy using both proprietary (OpenAI) and open-source (Mistral) LLMs. Results show Textonomy achieves competitive or superior alignment with human-annotated ground-truth clusters while reducing computational costs by over 99% compared to TopicGPT. Our work demonstrates that classification-based frameworks can effectively solve common topic modeling tasks, offering a scalable path to highly interpretable, goal-driven content analysis.

1 Introduction

The central goal of automated content analysis is to distill large volumes of text into meaningful, thematic categories (Stemler, 2000). Topic models have long been a primary tool for this task (Hoyle et al., 2022), yet they present a trilemma for practitioners, forcing a trade-off between interpretability, scalability, and accessibility. Traditional models like Latent Dirichlet Allocation (LDA) (Blei et al., 2003) are scalable but produce topics as word distributions that require significant human interpretation and are not necessarily aligned with the research question the human analyst has in mind (Hoyle et al., 2021; Chang et al., 2009). Users often have specific research questions or goals, which require topic models to produce not just coherent clusters of words, but meaningful, interpretable, and task-relevant categories (Stammbach et al., 2023; Wang et al., 2023; Doogan and Buntine, 2021; Hoyle et al., 2022). Modern neural methods like BERTopic (Grootendorst, 2022) improve topic coherence but still require post-hoc interpretation and, like many recent methods, assign a single topic per document by default.

Large Language Models (LLMs) have emerged as a powerful solution for interpretability, capable of generating human-readable topic labels and descriptions (Pham et al., 2024). However, methods like TopicGPT, which apply LLMs to every document, face prohibitive computational costs and long runtimes, limiting their scalability. Li et al. (2025) also highlight the expense of similar LLM-heavy approaches. Furthermore, many state-of-the-art approaches rely on proprietary, closed-source LLMs, raising concerns about reproducibility and accessibility for the wider research community (Li et al., 2025).

This paper addresses this trilemma by adapting and rigorously evaluating an implementation of the two-stage TnT-LLM framework by Wan et al. (2024). The two phases of the TnT-LLM framework are as follows.

1. **Taxonomy Generation:** An LLM iteratively creates a data-driven taxonomy from a small sample of document summaries, guided by a user-given analytical goal.

2. **LLM-Augmented Text Classification:** A subset of texts is pseudo-labeled by an LLM using the generated taxonomy, and this data is used to train a lightweight, efficient text classifier for large-scale inference.

[1] Code at https://github.com/xxx/textonomy

We present an implementation of this framework, which we call Textonomy. Using this two-phased taxonomy-based classification framework Textonomy aims to automate a form of emergent, goal-driven content analysis (Stemler, 2000) and is evaluated against common topic modeling methods. Our work thereby moves beyond the initial validation of the TnT-LLM framework on proprietary chat data. We position and evaluate Textonomy as a task-oriented alternative to topic models for the common goal of thematic document categorization. Our contributions are as follows.

1. We are the first to open source an implementation of and evaluate the TnT-LLM paradigm for general-purpose content analysis, comparing it against a full spectrum of topic modeling baselines.

2. We conduct this evaluation across two distinct domains using open source datasets (Wikipedia articles and US Congressional bills) to test for generalizability and mitigate risks of information leakage from LLM pretraining data.

3. We directly address reproducibility concerns by implementing and evaluating Textonomy with both proprietary (OpenAI) and open-source (Mistral) LLMs.

4. We demonstrate that this two-stage approach effectively navigates the interpretability-scalability trade-off, achieving competitive performance while being orders of magnitude more efficient than pure-LLM methods.

2 Related Work

Topic model evaluation has long been debated, with a push towards use-case-dependent metrics and human judgment alignment (Hoyle et al., 2021; Chang et al., 2009). Coherence measures like C_{NPMI} (Bouma, 2009; Aletras and Stevenson, 2013) and C_V (Röder et al., 2015) aim to proxy human interpretability, but their applicability to Neural Topic Models (NTMs) is debated and a substantial standardization gap was revealed in the topic modeling literature (Hoyle et al., 2021; Doogan and Buntine, 2021). Hoyle et al. (2022) advocate evaluating topic models based on criteria for "good" content analysis: reproducibility (alignment with human coding) and stability (intra-model consistency), reflecting inter-rater reliability and intra-rater reliability in traditional manual content analysis (Stemler, 2000).

Traditional topic models include Latent Dirichlet Allocation (LDA) (Blei et al., 2003), a foundational Bayesian probabilistic model which is a classic baseline in topic modeling. Among a wide range of different topic modeling methods (Abdelrazek et al., 2023) BERTopic (Grootendorst, 2022) stands out as a popular implementation of a Neural Topic Modeling by Clustering Embeddings (NTM-CE) that boasts good scalability combined with high topic coherence scores (Grootendorst, 2022).

Recent advances involve LLMs. TopicGPT (Pham et al., 2024) uses iterative LLM prompting for topic generation and assignment, yielding interpretable topics but at high computational cost. Li et al. (2025) compared several LLM-based methods, confirming high costs for models such as TopicGPT and LLooM (Lam et al., 2024). GoalEx (Wang et al., 2023) also uses LLMs but its individual topic assignment scales poorly. Query-driven models like Fang et al. (2021) allow topic specificity but lack general goal-orientation for the entire codeset.

The TnT-LLM framework (Wan et al., 2024), upon which Textonomy is built, was initially tested to generate user intent taxonomies from proprietary chat data, not directly for topic modeling or compared against topic models. Textonomy is, to our knowledge, the first application and evaluation of a TnT-LLM-based method for general-purpose topic modeling, focusing on balancing interpretability with scalability. Our work differentiates from Li et al. (2025) by focusing on the two-stage approach of TnT-LLM (LLM for taxonomy, lightweight classifier for scale) rather than a human-in-the-loop LLM-based system.

3 Textonomy: A TnT-LLM Approach

Textonomy implements the TnT-LLM framework (Wan et al., 2024) for scalable topic modeling or, more generally, automated content analysis. It consists of two main phases: Taxonomy Generation and LLM-Augmented Text Classification, with an overview given in Figure 1 and Figure 2, respectively. For its LLM components, Textonomy is designed for both capability and efficiency.

3.1 Phase 1: Taxonomy Generation

This phase creates a topic taxonomy tailored to the input data and a user-specified use case. By default,

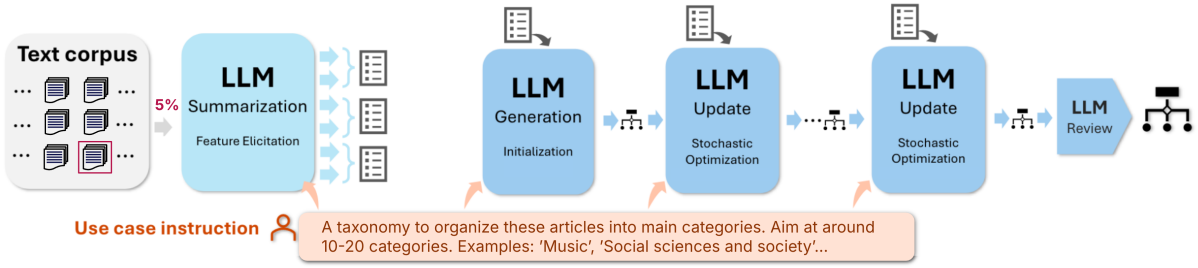

Figure 1: Conceptual overview of Phase 1 in Textonomy: Summaries from a data sample are batched for iterative taxonomy generation by an LLM. (2) The final taxonomy is used by an LLM to pseudo-label a training subset, upon which a lightweight classifier is trained for scalable inference. Figure adapted from Wan et al. (2024).

a 5% random sample of the input documents is used.

Stage 1: Summarization. Each document in the taxonomy sample is individually summarized by an LLM (default: GPT-4o-mini). The prompt requests a concise summary (e.g., 20 words) and a brief explanation (e.g., 30 words) for the summary, considering the use case input by the user. This step acts as a feature extraction process, distilling salient information relevant to the task.

Stage 2: Taxonomy Initialization, Updates, and Review The generated summaries are divided into equal-sized mini-batches. An LLM (default: o3-mini, selected for strong reasoning on such tasks) then performs a multi-stage reasoning process:

1. **Initialization:** The first batch of summaries is used to generate an initial taxonomy.
2. **Iterative Updates:** For subsequent batches, the LLM reviews the current taxonomy, rates its quality against predefined criteria (e.g., clarity, no overlap, relevance to use case), explains its rating, suggests edits based on the new batch of summaries, and provides an updated taxonomy.
3. **Final Review:** After processing all batches, the LLM performs a final review of the taxonomy without new data to ensure coherence and quality.

The prompts ensure the LLM adheres to format requirements (e.g., label structure with name and description, maximum number of categories) and quality criteria (e.g., mutual exclusivity, conciseness, accuracy). Users can adjust hyperparameters like category name/description length.

3.2 Phase 2: LLM-Augmented Text Classification

This phase scales up the classification using the generated taxonomy. A subset of the full dataset (default: 10%) is sampled for pseudo-labeling. An LLM (default: GPT-4o-mini) classifies these documents based on the final taxonomy from Phase 1. The prompt includes the document text and the full taxonomy (category names and descriptions).

This LLM-pseudo-labeled dataset is then used to train a lightweight text classifier. We use logistic regression by default, trained on sentence embeddings (default: `all-MiniLM-L6-v2`[2] via `sentence-transformers` (Reimers and Gurevych, 2019)). The resulting classifier can then efficiently categorize the entire corpus or new, unseen documents.

4 Experiments

4.1 Dataset and Preprocessing

To test generalizability and mitigate information leakage, we evaluated two distinct datasets, using the specific versions prepared and provided by Pham et al. (2024).

WikiText-103 (Wiki) This dataset (Merity et al., 2017) comprises 22,314 high-quality Wikipedia articles with 15 human-annotated high-level topic labels, which serve as our ground truth. The dataset is split into a training set of 14,290 articles and a test set of 8,024 articles.

US Congressional Bills (Bills) This dataset (Adler and Wilkerson, 2018) contains 32,661 bill summaries from the 110-114th U.S. Congresses. It is annotated with 21 high-level policy area labels

[2] https://huggingface.co/sentence-transformers/all-MiniLM-L6-v2

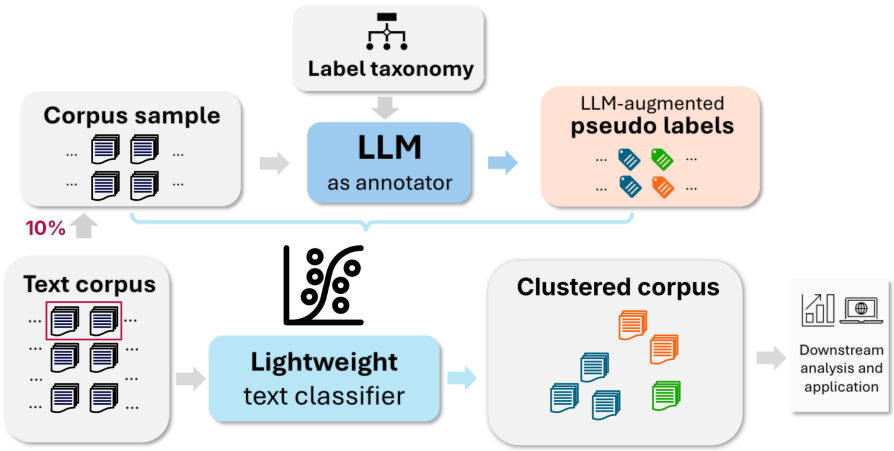

Figure 2: Conceptual overview of Phase 2 in Textonomy: The final taxonomy from Phase 1 is used by an LLM to pseudo-label a training subset, upon which a lightweight classifier is trained for scalable inference. Figure adapted from Wan et al. (2024).

that are used as our ground truth. Following the original authors' split, the dataset is divided into a training set of 17,419 summaries and a test set of 15,242 summaries.

Both datasets include preprocessed versions of the texts, featuring SpaCy tokenization, no lemmatization or stemming, and frequency-based word filtering. For our experiments with Textonomy, where less than the available number of training samples are required, we use a random subsample from the official training set of each respective corpus and document the seed for reproducibility.

4.2 Baselines

We compare Textonomy against:

- **LDA** (Blei et al., 2003): Implemented via Gensim (Řehůřek and Sojka, 2010).
- **BERTopic** (Grootendorst, 2022): Using default settings with `all-MiniLM-L6-v2` embeddings.
- **TopicGPT** (Pham et al., 2024): As results are expensive to reproduce, we report scores from their paper for the Wiki dataset where applicable and use their setup as a reference for our Textonomy experiments to create comparability.

For LDA and BERTopic, hyperparameters were kept to their respective libraries' defaults, with the exception of the number of topics (k), which was guided by Textonomy's output for a fair comparison. We acknowledge that our LDA implementation uses Gensim's variational inference, which may yield different results from a Gibbs sampler like Mallet, a point we address in our Limitations.

4.3 Evaluation Metrics

Definitions of all used evaluation metrics are detailed in Appendix A.

Topical Alignment (Interpretability): We measure alignment with the 15 ground-truth Wikipedia categories using:

- P_1: Harmonic mean of Purity and Inverse Purity (Zhao, 2005; Amigó et al., 2009).
- **ARI**: Adjusted Rand Index (Hubert and Arabie, 1985).
- **NMI**: Normalized Mutual Information (Strehl and Ghosh, 2003).

We selected P_1, ARI, and NMI as our primary external cluster metrics to measure topical alignment due to their complementary strengths and to create comparability to TopicGPT (Pham et al., 2024). P_1 balances the purity of topics (are documents in a topic from one class?) and the completeness of classes (are documents from a class in one topic?). The ARI assesses the similarity between two clusterings while accounting for agreements that could occur by chance. It is particularly sensitive to differences in the underlying structure of the clusterings because it performs pairwise checks to see if items are grouped together consistently. NMI, an information-theoretic measure, quantifies the mutual dependence between the model's clustering and true classes, handling differing numbers

of clusters well and offering insights into shared information.

Internal Quality Metrics:

- **Coherence:** C_{NPMI} (Aletras and Stevenson, 2013) and C_V (Röder et al., 2015), calculated on the keyword representation produced by LDA and c-TF-IDF based keywords for BERTopic and Textonomy.
- **Diversity:** Pairwise Jaccard Distance (D_{PJD}) (Tran et al., 2013) and Proportion of Unique Words (D_{PUW}) (Dieng et al., 2020).
- **Validity:** Outlier Ratio (U_{OR}) and an LLM-based usefulness score (U_{LLM}) assessing relevance, clarity, comprehensiveness, and distribution against the user-defined purpose.

Stability: We assess stability by comparing topic assignments from different runs of Textonomy (with variations in data, prompts, or LLM settings) using P_1, ARI, and NMI against a default Textonomy run. LDA stability (average over 10 runs) serves as a baseline.

4.4 Textonomy Configuration

We evaluate two primary configurations of Textonomy:

- **Textonomy (Default):** Uses OpenAI's 'o3-mini' for taxonomy generation and 'GPT-4o-mini' for summarization and pseudo-labeling. This configuration prioritizes performance based on powerful proprietary models. OpenAI's o3-mini was selected for taxonomy generation due to its strong performance on generating taxonomies fitting the summaries batch. In contrast, non-reasoning models like GPT-4o show tendency to create more generic taxonomies. For the less reasoning-intensive tasks of document summarization and pseudo-labeling, GPT-4o-mini was chosen for its balance of good performance and significantly lower operational cost compared to larger flagship models.
- **Textonomy (OSS):** Uses 'Mistral-Large-Instruct-2411'[3] for all LLM tasks to evaluate the viability of a fully open source pipeline. This model is open source and free to use under the Mistral AI Non-Production License[4].

For both configurations, the user-defined use case for the Wiki dataset was: "A taxonomy to organise these articles into main categories. Aim at around 10-20 categories. Examples: 'Music', 'Social sciences and society'...". For both Textonomy models on the Wiki dataset, Phase 1 used a sample of 500 documents, and Phase 2 used 1,340 documents for classifier training. For a detailed set of all hyperparameters used, refer to Appendix B.

5 Results and Discussion

We present results for interpretability (topical alignment), internal quality, stability, and computational complexity. For Textonomy, LDA, and BERTopic, we report the average over three runs on the test sets, alongside the best run. For TopicGPT, we refer to published results (Pham et al., 2024).

5.1 Interpretability and Topic Quality

Tables 1 and 2 show the topical alignment and internal quality metrics for the Wiki and Bills datasets, respectively.

On the **Wiki dataset**, Textonomy consistently performs well in topical alignment. Its average P_1 score (0.73) matches TopicGPT's best run at default settings (0.73) and is close to its refined version (0.74). Textonomy significantly outperforms TopicGPT on ARI (average 0.68 vs. TopicGPT's 0.58-0.60), indicating better structural agreement with ground-truth clusters. This ARI score approaches levels indicative of good inter-rater reliability (e.g., in comparison to the suggested level for Cohen's Kappa ≈ 0.8 by Stemler (2000)). On NMI, Textonomy (0.66) is comparable to LDA (0.66) but slightly below TopicGPT (0.70-0.71), potentially due to the classifier's handling of imbalanced or smaller classes from the taxonomy. Both LLM-based methods (Textonomy and TopicGPT) generally surpass LDA and substantially outperform BERTopic on alignment metrics. BERTopic's low alignment scores, despite reasonable coherence, highlight the known gap between some automated metrics and human-like clustering for NTMs (Hoyle et al., 2022).

For internal metrics, Textonomy achieves coherence (C_{NPMI}, C_V) and diversity (D_{PJD}, D_{PUW}) scores competitive with LDA and BERTopic. Its

[3] https://huggingface.co/mistralai/Mistral-Large-Instruct-2411

[4] https://mistral.ai/static/licenses/MRL-0.1.md

Model	Run Type	Alignment			Coherence		Diversity		Usefulness		k
		$P_1 \uparrow$	ARI $\uparrow$	NMI $\uparrow$	$C_{NPMI} \uparrow$	$C_V \uparrow$	$D_{PJD} \uparrow$	$D_{PUW} \uparrow$	$U_{OR} \downarrow$	$U_{LLM} \uparrow$	
Ground Truth	Avg	1.00	1.00	1.00	0.12	0.67	0.99	0.87	0.00	0.85	15
Textonomy	Best	**0.74**	**0.70**	0.67	**0.11**	0.64	0.98	0.84	**0.00**	**0.90**	16
	Avg	**0.73 ± 0.01**	**0.68 ± 0.01**	**0.66 ± 0.01**	**0.11 ± 0.00**	**0.64 ± 0.01**	0.98 ± 0.00	**0.82 ± 0.02**	**0.00 ± 0.00**	**0.88 ± 0.02**	14.7
Textonomy (OSS)	Best	0.70	0.56	0.64	**0.11**	**0.65**	**0.99**	**0.89**	0.02	0.75	13
	Avg	0.66 ± 0.03	0.56 ± 0.00	0.61 ± 0.02	**0.11 ± 0.01**	0.63 ± 0.02	**0.99 ± 0.00**	**0.85 ± 0.03**	**0.01 ± 0.01**	**0.73 ± 0.02**	16.3
TopicGPT	Default	0.73	0.58	**0.71**	-	-	-	-	-	-	31
	Refined	**0.74**	0.60	0.70	-	-	-	-	-	-	22
LDA	Best	0.68	0.59	0.66	0.10	0.61	0.98	**0.85**	**0.00**	0.60	13
	Avg	0.66 ± 0.01	0.53 ± 0.03	**0.66 ± 0.00**	0.10 ± 0.00	0.62 ± 0.01	0.98 ± 0.00	0.81 ± 0.02	**0.00 ± 0.00**	0.69 ± 0.04	14.7
BERTopic	Best	0.49	0.15	0.40	**0.11**	0.62	0.98	0.79	0.39	0.70	16
	Avg	0.46 ± 0.03	0.12 ± 0.02	0.37 ± 0.03	0.10 ± 0.00	0.63 ± 0.00	0.98 ± 0.00	0.80 ± 0.01	0.43 ± 0.03	0.73 ± 0.02	14.7

Table 1: Interpretability and internal quality results on the **Wiki** test dataset. Alignment scores compare model clusters to 15 human-annotated ground-truth categories. Higher is better for all metrics except U_{OR}. Best scores per metric and run type are bolded. TopicGPT results from Pham et al. (2024). k is number of topics.

Model	Run Type	Alignment			Coherence		Diversity		Usefulness		k
		$P_1 \uparrow$	ARI $\uparrow$	NMI $\uparrow$	$C_{NPMI} \uparrow$	$C_V \uparrow$	$D_{PJD} \uparrow$	$D_{PUW} \uparrow$	$U_{OR} \downarrow$	$U_{LLM} \uparrow$	
Ground Truth	Avg	1	1	1	0.12	0.67	0.99	0.87	0	0.85	21
Textonomy	Best	0.50	0.34	0.42	0.06	0.49	0.89	0.57	**0.00**	**0.95**	25
	Avg	0.47 ± 0.02	0.31 ± 0.02	**0.39 ± 0.02**	**0.06 ± 0.01**	0.47 ± 0.01	0.88 ± 0.01	0.56 ± 0.02	**0.00 ± 0.00**	**0.88 ± 0.05**	23.3
Textonomy (OSS)	Best	0.56	0.39	0.43	0.05	0.47	0.86	0.55	**0.00**	0.75	19
	Avg	**0.50 ± 0.05**	**0.32 ± 0.05**	**0.39 ± 0.03**	0.05 ± 0.01	0.47 ± 0.01	0.86 ± 0.00	0.55 ± 0.00	**0.00 ± 0.00**	0.75 ± 0.00	18.33
TopicGPT	Default	**0.57**	**0.42**	**0.51**	-	-	-	-	-	-	94
	Refined	**0.57**	0.40	0.49	-	-	-	-	-	-	24
LDA	Best	0.45	0.23	0.36	**0.07**	**0.52**	**0.95**	**0.70**	**0.00**	0.85	23
	Avg	0.43 ± 0.01	0.24 ± 0.00	0.34 ± 0.01	**0.06 ± 0.00**	**0.51 ± 0.00**	**0.94 ± 0.00**	**0.69 ± 0.01**	**0.00 ± 0.00**	0.80 ± 0.06	23.3
BERTopic	Best	0.29	0.05	0.22	0.05	0.46	0.91	0.60	0.42	0.70	25
	Avg	0.28 ± 0.01	0.05 ± 0.00	0.21 ± 0.00	0.04 ± 0.01	0.44 ± 0.01	0.91 ± 0.00	0.64 ± 0.03	0.43 ± 0.01	0.71 ± 0.03	23.3

Table 2: Interpretability and internal quality results on the **Bills** test dataset. Alignment scores compare model clusters to 21 human-annotated ground-truth categories. Higher is better for all metrics except U_{OR}. Best scores per metric and run type are bolded. TopicGPT results from Pham et al. (2024). k is number of topics.

outlier ratio (U_{OR}) is 0, ideal for this dataset where all articles are labeled. Textonomy also scores highest on LLM-evaluated usefulness (U_{LLM}), likely benefiting from its generation of descriptive category names and descriptions compared to keyword lists from LDA/BERTopic.

On the **Bills dataset**, a more specialized domain less likely to be affected by LLM pre-training data memorization, the results show a different dynamic. Here, TopicGPT maintains a lead in alignment metrics (P_1, ARI, NMI), likely benefiting from its per-document LLM analysis on this complex textual data. However, both the proprietary and open-source (OSS) versions of Textonomy significantly outperform the traditional LDA and BERTopic baselines on all alignment scores. Notably, Textonomy (OSS) with Mistral achieves the best performance of the two Textonomy variants on P_1 and ARI, demonstrating that open-source models can be highly effective and even surpass their proprietary counterparts in specific domains. In contrast to the Wiki results, LDA shows stronger performance on internal coherence (C_{NPMI}, C_V) and diversity (D_{PJD}, D_{PUW}) metrics than all other models on this dataset, suggesting its keyword-based topics are statistically distinct, even if they align less well with the human-annotated ground truth. Once again, Textonomy stands out with a perfect outlier ratio ($U_{OR} = 0.00$) and high usefulness scores (U_{LLM}), reinforcing its ability to produce comprehensive and interpretable categorizations.

5.2 Stability

Table 3 presents Textonomy's stability under various perturbations, compared to an LDA baseline.

Method	Setting Variation	$P_1 \uparrow$	ARI $\uparrow$	NMI $\uparrow$
LDA	Default ($k = 15$, avg. 10 runs)	0.75	0.66	**0.76**
Textonomy	Default settings ($k = 15$)	**0.80**	**0.78**	0.73
	Shuffled training data ($k = 16$)	**0.82**	**0.79**	0.75
	Different training data ($k = 15$)	**0.77**	**0.75**	0.71
	Generic use case prompt ($k = 12$)	**0.81**	**0.76**	0.74
	Assign w. GPT-4o ($k = 18$)	0.72	**0.71**	0.70

Table 3: Stability of topic assignments for Textonomy and LDA on the Wiki dataset. Metrics compare assignments from varied settings against a default Textonomy run. Higher scores indicate greater stability.

Textonomy demonstrates high stability, generally meeting or exceeding the LDA baseline. Mi-

nor changes like data shuffling or using a generic prompt have a minor impact, with ARI scores around 0.76-0.79. This suggests robust "intra-coder reliability". More substantial changes, like using completely different training data or a different LLM for pseudo-labeling (GPT-4o instead of GPT-4o-mini), result in slightly larger deviations but maintain reasonably high agreement. This indicates that Textonomy's two-phase process effectively dampens variance from LLM non-determinism and training data choice.

5.3 Computational Complexity

A key advantage of Textonomy is its efficiency. On the Wiki dataset (8,024 test documents, with preceding training/taxonomy generation steps), Textonomy took, averaged over 3 runs, approximately 6.8 minutes per run. In contrast, TopicGPT is estimated to take around 7.5 hours (450 minutes) for a similar task (Li et al., 2025; Pham et al., 2024). This represents a ∼66x speed-up or a 98.5% reduction in time.

Textonomy using an open source Mistral model (Mistral-Large-Instruct-2411 for all LLM tasks) did not incur any cost, since the free-to-use API Endpoint[5] provided by Mistral was used. If the experiment was to be reproduced without the Mistral API, note that a self-hosted version of Mistral-Large-Instruct-2411 would probably incur some hosting costs but would heavily depend on how the setup is realized and is thus hard to estimate.

Monetarily, for the experimental setup described, a single Textonomy run cost approximately $0.93 using OpenAI API calls (GPT-4o-mini for summarization/classification, o3-mini for taxonomy). TopicGPT, as reported by Pham et al. (2024), cost $155 for the Wiki dataset (though prices and models may have changed, our re-estimation with current models like GPT-4/GPT-3.5-turbo suggests costs around $150, or $10.1 with Textonomy's cheaper LLMs). Textonomy's default configuration thus achieves a cost reduction of over 99.4% compared to the original TopicGPT, and remains at least 11x cheaper even if TopicGPT were to use the same more economical LLMs as Textonomy.

The costs of running LDA and BERTopic locally on a CPU, without any required external API or beyond average compute power, are considered negligible in comparison. Regarding runtime, BERTopic is the fastest (avg. 2.3 min), and LDA is comparable to Textonomy in time (avg. 7.1 min).

[5] https://mistral.ai/news/september-24-release

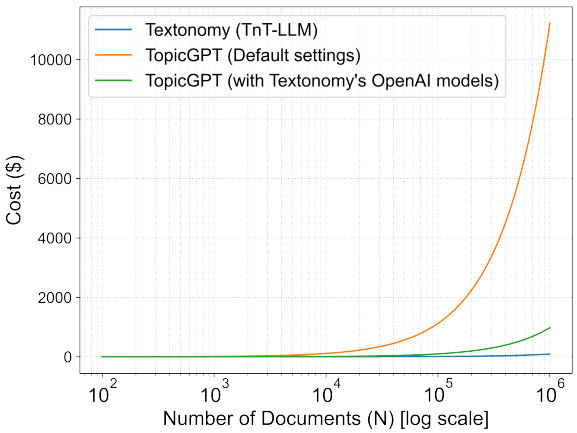

Figure 3: Estimated cost for Textonomy, TopicGPT and TopicGPT with the OpenAI models used in Textonomy, assuming an average document length like in the Wiki dataset of 2500 words.

The substantial cost and time savings are due to Textonomy's design: LLMs are used sparingly on smaller data samples for taxonomy creation and pseudo-labeling, while large-scale inference relies on an efficient lightweight classifier.

The dramatic reductions in time and cost achieved by Textonomy are not merely incremental improvements but fundamentally alter the feasibility of using advanced LLM capabilities for topic modeling at scale (see Figure 3). This efficiency opens doors for analyzing much larger datasets, conducting more extensive hyperparameter exploration, or deploying topic modeling in resource-constrained environments where methods like TopicGPT would be prohibitive. It allows researchers to iterate faster and apply sophisticated analysis to corpora that were previously intractable with such methods.

5.4 Text Classifier Performance

The lightweight logistic regression classifier in Textonomy achieved an average F1 score of 79% over topical alignment experiments on both datasets when evaluated on its ability to reproduce the LLM's pseudo-labels on a 25% held-out test set from the 1,340 sampled documents. Textonomy with open source LLMs achieved an average F1 score of 73%. While this indicates strong mimicry of the LLM's decisions, any errors made by the LLM during pseudo-labeling could propagate. However, given that LLM errors in zero-shot classification often result in assignment to semantically related (though not identical) categories, the

impact on overall topic coherence and interpretability might be less severe than random errors. The high topical alignment scores (Table 1) despite this two-step process (LLM pseudo-labeling, then classifier training) suggest the overall approach is effective. Future work could explore more advanced distillation or prompting techniques to further enhance classifier accuracy.

5.5 Insights from Qualitative Comparison to Ground Truth

Qualitative analysis of Textonomy's generated taxonomy against the 15 Wikipedia Supercategories revealed that Textonomy produced topics that are largely in agreement with ground-truth labels (see Figure 5 in Appendix C), while making some reasonable adjustments adequate given the training data. Many generated topics showed strong semantic overlap with ground-truth categories (e.g., "Video Games," "Music & Pop" for "Music," "Military History" for "Warfare"). Some differences arose where Textonomy created more granular topics based on data prevalence. For example, the majority of documents from the ground-truth class "Engineering and technology" are about highways, airports, etc., which is more of a problem with the training/test dataset than with Textonomy, which created a cluster for these documents called "Transport & Urban" (see Figure 4 in Appendix C for the contingency matrix used for comparison). Differences arose also where the taxonomy generation sample had sparse representation of certain ground-truth categories, impacting NMI scores for those underrepresented classes in the final classification. This highlights the importance of the taxonomy sampling step. Such data-driven distinctions can be beneficial for exploratory analysis but also highlight the influence of the taxonomy generation sample. If this sample underrepresents certain ground-truth categories or presents a skewed view, the resulting taxonomy will reflect that, potentially impacting metrics like NMI if the test set has a different distribution.

6 Conclusion

This paper introduced Textonomy, an implementation of the TnT-LLM framework for automated task-oriented content analysis that prioritizes scalability, interpretability, and cost-efficiency. Our experiments on the WikiText-103 and Bills datasets demonstrate that Textonomy achieves topical alignment and stability competitive with or exceeding state-of-the-art topic modeling baselines, including the prompt-based framework TopicGPT. Notably, Textonomy achieves these results while reducing computational costs by over 99% and runtime by over 98% compared to TopicGPT.

Model	Avg. Runtime (min)	Est. API Cost (USD)
Textonomy	6.8	$0.93
Textonomy (OSS)	57.6	$0.00 (free API)
TopicGPT (est.)	∼450.0	∼$150.00
LDA	7.1	$0.00 (local)
BERTopic	2.3	$0.00 (local)

Table 4: Computational complexity on the Wiki dataset. Runtimes are averages from our experiments. Textonomy (OSS) would incur self-hosting costs if not using a free API endpoint. Cost of running LDA and BERTopic on a local CPU are considered negligible in comparison.

Additionally, a version of Textonomy using an open-source LLM by Mistral was introduced and evaluated to address concerns about reliance on proprietary models. This variant is highly viable: while competitive on Wiki, it surpassed the proprietary version on specialized Bills data. This demonstrates that open-source models can be exceptionally effective for domain-specific tasks and confirms that the robustness of the Textonomy framework is not solely dependent on a specific proprietary LLM, paving the way for building fully transparent, accessible, and low-cost tools for content analysis.

Textonomy's two-phase approach—LLM-driven iterative taxonomy generation on summaries, followed by training a lightweight classifier on LLM pseudo-labels—effectively balances the nuanced reasoning capabilities of LLMs with the need for efficient large-scale processing. This makes it a viable solution for practical automated text content analysis in large corpora.

Future work could explore Textonomy's application to diverse domains and languages, investigate adaptive hyperparameter tuning, further refine the LLM-augmented classification stage, and explore the generation of hierarchical taxonomies. Textonomy offers a significant step towards making advanced, LLM-enhanced topic modeling more accessible and practical for a wider range of research and application scenarios.

All codes and data are made publicly available at [link omitted for review] to facilitate reproducibility and further research [similar to the provided code.zip and data.zip for review].

Limitations

The findings of this study are subject to several limitations:

- **Dataset Specificity:** Results on Wikipedia (WikiText-103) and Bills may not generalize to all domains, especially those with highly specialized jargon or data not well-represented in LLM pre-training corpora. In these cases, the used LLMs might be fine-tuned on the used dataset, which would increase the cost and effort involved in training Textonomy.
- **Information Leakage:** The potential for data memorization by LLMs on a well-known dataset like Wikipedia is a concern, although Textonomy's batch-based reasoning for taxonomy and use of a separate classifier may mitigate direct memorization effects compared to per-document LLM prompting. On the other side, the Textonomy did not yield a perfect score nor did it perfectly recall the ground-truth labels. Additionally, on the Bills dataset, the labels are not directly associated with the texts, rather being available in a separate file[6] where the numerical code of the label points to the URL of the Bill (Pham et al., 2024).
- **Evaluation Metrics:** While we use established metrics, topic model evaluation remains complex. Ground-truth alignment is valuable but does not capture all aspects of "good" content analysis. Additionally, using external cluster metrics to measure topical alignment and stability assumes one-to-one mappings between documents and topic clusters, where in classic topic modeling evaluation the assumption is a one-to-many relationship. This setup, while common for benchmarking, does not assess the ability of models like LDA to capture a document's topic mixture. Internal metrics may not always correlate with human judgment for LLM-generated topics.
- **LLM Dependencies:** Performance of Textonomy with proprietary LLMs relies on the OpenAI API. This involves costs, potential API changes, and lack of full transparency into model architecture and training data, which can perpetuate biases (Bender et al., 2021). Even when using Textonomy (OSS) with open-source LLMs, some general drawbacks of LLMs apply, like resource intensiveness and nondeterministic behavior. While Textonomy aims for interpretability in its outputs, the internal workings of the used LLMs remain a black box. Furthermore, this work does not conduct a formal analysis of biases (e.g., as defined by Blodgett et al. (2020)) within the LLMs or Textonomy's final outputs, but acknowledges the risk of bias propagation from the pre-trained LLMs used in summarization, taxonomy generation, and pseudo-labeling. A core assumption is that the two-phase process—taxonomy generation from summaries and subsequent classification—effectively captures the corpus's essential thematic structure without critical information loss compared to methods that might use full documents for every LLM interaction. Violations of this, e.g., if summaries miss crucial nuances for specific topics, could impact taxonomy quality.
- **Language:** Evaluation was limited to English.
- **Hyperparameter Sensitivity:** While Textonomy shows stability, optimal performance for the taxonomy generation phase (e.g., sample size, batching strategy) might require some tuning depending on dataset characteristics and desired granularity, which was not exhaustively explored.
- **Areas of Improvement:** Some opportunities for future improvements and investigation into the validity of this work include, although are not limited to: Our LDA baseline uses Gensim's variational inference and may underperform a Gibbs sampling implementation like Mallet. Finally, we did not perform extensive ablations on internal hyperparameters like summary length or the number of pseudo-labels, which we leave for future work.

[6] http://www.congressionalbills.org/download.html

Acknowledgments

[omitted for review]

References

Aly Abdelrazek, Yomna Eid, Eman Gawish, Walaa Medhat, and Ahmed Hassan. 2023. Topic modeling al-

gorithms and applications: A survey. *Information Systems*, 112:102131.

E. Scott Adler and John Wilkerson. 2018. Congressional Bills Project: 1995-2018.

Nikolaos Aletras and Mark Stevenson. 2013. Evaluating Topic Coherence Using Distributional Semantics. In *Proceedings of the 10th International Conference on Computational Semantics (IWCS 2013) – Long Papers*, pages 13–22, Potsdam, Germany. Association for Computational Linguistics.

Enrique Amigó, Julio Gonzalo, Javier Artiles, and Felisa Verdejo. 2009. A comparison of extrinsic clustering evaluation metrics based on formal constraints. *Information Retrieval*, 12(4):461–486.

Emily M. Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. 2021. On the Dangers of Stochastic Parrots: Can Language Models Be Too Big? In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '21, pages 610–623, New York, NY, USA. Association for Computing Machinery.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent dirichlet allocation. *The Journal of Machine Learning Research*, 3:993–1022.

Su Lin Blodgett, Solon Barocas, Hal Daumé III, and Hanna Wallach. 2020. Language (Technology) is Power: A Critical Survey of "Bias" in NLP. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5454–5476, Online. Association for Computational Linguistics.

G. Bouma. 2009. Normalized (pointwise) mutual information in collocation extraction.

Jonathan Chang, Sean Gerrish, Chong Wang, Jordan Boyd-graber, and David Blei. 2009. Reading Tea Leaves: How Humans Interpret Topic Models. In *Advances in Neural Information Processing Systems*, volume 22. Curran Associates, Inc.

Adji B. Dieng, Francisco J. R. Ruiz, and David M. Blei. 2019. Topic Modeling in Embedding Spaces. *arXiv preprint*. ArXiv:1907.04907 [cs, stat].

Adji B. Dieng, Francisco J. R. Ruiz, and David M. Blei. 2020. Topic Modeling in Embedding Spaces. *Transactions of the Association for Computational Linguistics*, 8:439–453.

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.

Zheng Fang, Yulan He, and Rob Procter. 2021. A Query-Driven Topic Model. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1764–1777, Online. Association for Computational Linguistics.

Maarten Grootendorst. 2022. BERTopic: Neural topic modeling with a class-based TF-IDF procedure. *arXiv preprint*. ArXiv:2203.05794 [cs].

Alexander Miserlis 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 Miserlis Hoyle, Rupak Sarkar, Pranav Goel, 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.

Lawrence Hubert and Phipps Arabie. 1985. Comparing partitions. *Journal of Classification*, 2(1):193–218.

Michelle S. Lam, Janice Teoh, James Landay, Jeffrey Heer, and Michael S. Bernstein. 2024. Concept Induction: Analyzing Unstructured Text with High-Level Concepts Using LLooM. In *Proceedings of the CHI Conference on Human Factors in Computing Systems*, pages 1–28. ArXiv:2404.12259 [cs].

Zongxia Li, Lorena Calvo-Bartolomé, Alexander Hoyle, Paiheng Xu, Alden Dima, Juan Francisco Fung, and Jordan Boyd-Graber. 2025. Large Language Models Struggle to Describe the Haystack without Human Help: Human-in-the-loop Evaluation of LLMs. *arXiv preprint*. ArXiv:2502.14748 [cs].

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. 2017. Pointer Sentinel Mixture Models.

Chau Pham, Alexander Hoyle, Simeng Sun, Philip Resnik, and Mohit Iyyer. 2024. TopicGPT: A Prompt-based Topic Modeling Framework. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 2956–2984, Mexico City, Mexico. Association for Computational Linguistics.

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.

Michael Röder, Andreas Both, and Alexander Hinneburg. 2015. Exploring the Space of Topic Coherence Measures. In *Proceedings of the Eighth ACM International Conference on Web Search and Data Mining*, pages 399–408, Shanghai China. ACM.

C. E. Shannon. 1948. A mathematical theory of communication. *The Bell System Technical Journal*, 27(3):379–423. Conference Name: The Bell System Technical Journal.

Dominik Stammbach, Vilém Zouhar, Alexander Hoyle, Mrinmaya Sachan, and Elliott Ash. 2023. Revisiting Automated Topic Model Evaluation with Large Language Models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9348–9357, Singapore. Association for Computational Linguistics.

Steve Stemler. 2000. An overview of content analysis. *Practical Assessment, Research, and Evaluation*, 7(1). Number: 1 Publisher: University of Massachusetts Amherst Libraries.

Alexander Strehl and Joydeep Ghosh. 2003. Cluster ensembles — a knowledge reuse framework for combining multiple partitions. *J. Mach. Learn. Res.*, 3(null):583–617.

Nam Khanh Tran, Sergej Zerr, Kerstin Bischoff, Claudia Niederée, and Ralf Krestel. 2013. Topic Cropping: Leveraging Latent Topics for the Analysis of Small Corpora. In *Research and Advanced Technology for Digital Libraries*, pages 297–308, Berlin, Heidelberg. Springer.

Mengting Wan, Tara Safavi, Sujay Kumar Jauhar, Yujin Kim, Scott Counts, Jennifer Neville, Siddharth Suri, Chirag Shah, Ryen W. White, Longqi Yang, Reid Andersen, Georg Buscher, Dhruv Joshi, and Nagu Rangan. 2024. TnT-LLM: Text Mining at Scale with Large Language Models. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '24, pages 5836–5847, New York, NY, USA. Association for Computing Machinery.

Zihan Wang, Jingbo Shang, and Ruiqi Zhong. 2023. Goal-Driven Explainable Clustering via Language Descriptions. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 10626–10649, Singapore. Association for Computational Linguistics.

Xiaobao Wu, Thong Nguyen, and Anh Tuan Luu. 2024. A survey on neural topic models: methods, applications, and challenges. *Artificial Intelligence Review*, 57(2):18.

Ying Zhao. 2005. *Criterion functions for document clustering*. phd, University of Minnesota, USA. AAI3180039 ISBN-10: 0542203189.

Radim Řehůřek and Petr Sojka. 2010. Software Framework for Topic Modelling with Large Corpora. Pages: 45–50 Series: Proceedings of LREC 2010 workshop New Challenges for NLP Frameworks original-date: 2011-02-10T07:43:04Z.

A Evaluation Metrics

This appendix details the external and internal clustering metrics used to evaluate the topic models in this paper.

A.1 Common Notation

The following notation is used:

- Let $D = \{d_1, d_2, \ldots, d_N\}$ be our dataset containing N documents.
- Let $\mathcal{T} = \{T_1, T_2, \ldots, T_K\}$ be the set of K topic clusters obtained from the topic model, where each document $d_j \in D$ is assigned to exactly one topic cluster $T(d_j) \in \mathcal{T}$.
- Let $\mathcal{W} = \{W_1, W_2, \ldots, W_K\}$ be the set of K topic keyword representations obtained from the topic model or a separate class-based TF-IDF procedure, where:
  - each keyword representation W_k belongs to topic T_k of the same index k,
  - $W_k = \{w_1, w_2, \ldots, w_Q\}$ is the set of Q keywords chosen to represent topic T_k, and
  - Q is a hyperparameter that is set to 10 by default as in most works (Röder et al., 2015; Doogan and Buntine, 2021).
- Let $\mathcal{C} = \{C_1, C_2, \ldots, C_I\}$ be the set of I human-annotated ground-truth classes, where each document $d_j \in D$ belongs to exactly one class $C(d_j) \in \mathcal{C}$.
- Let $\mathbf{M}$ be the contingency matrix where:
  - $M_{i,k}$ is the number of documents assigned to both ground-truth class C_i and topic cluster T_k, i.e.
$$M_{i,k} = |\{d_j \in D \mid C(d_j) = C_i, T(d_j) = T_k\}|, \tag{1}$$
  - the row sums represent the total documents in each ground-truth class, i.e.
$$|C_i| = \sum_{k=1}^{K} M_{i,k}, \tag{2}$$
  - and the column sums represent the total documents in each topic cluster, i.e.
$$|T_k| = \sum_{i=1}^{I} M_{i,k}. \tag{3}$$

A.2 External Cluster Metrics

These metrics assess the agreement between model-generated topic clusters $\mathcal{T}$ and ground-truth classes $\mathcal{C}$. We used implementations from scikit-learn where available.

P_1: Harmonic Mean of Purity and Inverse Purity Purity measures the extent to which each topic T_k contains documents from primarily one class C_i. Inverse Purity measures the extent to which each class C_i is represented by a single topic T_k (Zhao, 2005).

$$\text{Purity}(\mathcal{T}, \mathcal{C}) = \frac{1}{N} \sum_{k=1}^{K} \max_i M_{i,k} \tag{4}$$

$$\text{Purity}^{-1}(\mathcal{T}, \mathcal{C}) = \frac{1}{N} \sum_{i=1}^{I} \max_k M_{i,k}. \tag{5}$$

P_1 is their harmonic mean, balancing both aspects (Amigó et al., 2009):

$$P_1(\mathcal{T}, \mathcal{C}) = 2 \times \frac{\text{Purity}(\mathcal{T}, \mathcal{C}) \times \text{Purity}^{-1}(\mathcal{T}, \mathcal{C})}{\text{Purity}(\mathcal{T}, \mathcal{C}) + \text{Purity}^{-1}(\mathcal{T}, \mathcal{C})}. \tag{6}$$

A P_1 score of 1 indicates perfect alignment.

ARI: Adjusted Rand Index The Rand Index (RI) (Rand, 1971) measures similarity between clusterings. The Adjusted Rand Index (ARI) (Hubert and Arabie, 1985) corrects RI for chance. The general form is:

$$\text{ARI}(\mathcal{T}, \mathcal{C}) = \frac{\text{RI} - \mathbb{E}[\text{RI}]}{\max(\text{RI}) - \mathbb{E}[\text{RI}]}. \tag{7}$$

Using the contingency matrix $\mathbf{M}$, ARI is calculated as:

$$\text{ARI}(\mathcal{T}, \mathcal{C}) = \frac{\sum_{i=1}^{I}\sum_{k=1}^{K}\binom{M_{i,k}}{2} - \frac{\left[\sum_{i=1}^{I}\binom{|C_i|}{2}\right]\left[\sum_{k=1}^{K}\binom{|T_k|}{2}\right]}{\binom{N}{2}}}{\frac{1}{2}\left[\sum_{i=1}^{I}\binom{|C_i|}{2} + \sum_{k=1}^{K}\binom{|T_k|}{2}\right] - \frac{\left[\sum_{i=1}^{I}\binom{|C_i|}{2}\right]\left[\sum_{k=1}^{K}\binom{|T_k|}{2}\right]}{\binom{N}{2}}}. \tag{8}$$

Substituting these into the general ARI formula gives the specific calculation used. ARI ranges from -1 to 1, where 1 is perfect agreement, 0 is random agreement.

NMI: Normalized Mutual Information NMI is an information-theoretic measure quantifying the mutual dependence between the topic clustering $\mathcal{T}$ and ground-truth classes $\mathcal{C}$ (Strehl and Ghosh, 2003). It normalizes Mutual Information (MI) by the average of their entropies.

The joint and marginal probabilities are defined as:

$$P(C_i, T_k) = \frac{M_{i,k}}{N}, \tag{9}$$

$$P(C_i) = \frac{|C_i|}{N}, \tag{10}$$

$$P(T_k) = \frac{|T_k|}{N}. \tag{11}$$

MI is defined as (Shannon, 1948):

$$I(\mathcal{T}; \mathcal{C}) = \sum_{i=1}^{I}\sum_{k=1}^{K} P(C_i, T_k) \log \frac{P(C_i, T_k)}{P(C_i)P(T_k)}. \tag{12}$$

This can also be written using counts from the contingency matrix $\mathbf{M}$:

$$I(\mathcal{T}; \mathcal{C}) = \sum_{i=1}^{I}\sum_{k=1}^{K} \frac{M_{i,k}}{N} \log \frac{M_{i,k}N}{|C_i||T_k|}. \tag{13}$$

The entropies of the topic clusters $\mathcal{T}$ and ground-truth classes $\mathcal{C}$ are:

$$H(\mathcal{T}) = -\sum_{k=1}^{K} P(T_k) \log P(T_k) = -\sum_{k=1}^{K} \frac{|T_k|}{N} \log \frac{|T_k|}{N}, \tag{14}$$

$$H(\mathcal{C}) = -\sum_{i=1}^{I} P(C_i) \log P(C_i) = -\sum_{i=1}^{I} \frac{|C_i|}{N} \log \frac{|C_i|}{N}. \tag{15}$$

NMI is then:

$$\text{NMI}(\mathcal{T}, \mathcal{C}) = \frac{2 \cdot I(\mathcal{T}; \mathcal{C})}{H(\mathcal{T}) + H(\mathcal{C})}. \tag{16}$$

NMI ranges from 0 (no mutual information) to 1 (perfect correlation).

A.3 Internal Cluster Metrics

These metrics assess topic quality based on the generated topics themselves, without reference to ground-truth labels.

A.3.1 Topic Coherence

Measures semantic similarity among high-scoring words within a topic.

C_{NPMI}: Normalized Pointwise Mutual Information Coherence NPMI measures the co-occurrence of two words w_i, w_j normalized by their joint probability, resulting in a score between -1 and 1 (Bouma, 2009).

$$\text{NPMI}(w_i, w_j) = \frac{\log \frac{P(w_i, w_j)}{P(w_i)P(w_j)}}{-\log P(w_i, w_j)} \tag{17}$$

$P(w_i, w_j)$ is the probability of w_i and w_j co-occurring (e.g., in a sliding window within a reference corpus), and $P(w_i), P(w_j)$ are their individual probabilities. The topic coherence C_{NPMI} is the average NPMI score over the top Q words for each topic, then averaged over all topics (Aletras and Stevenson, 2013). Higher values indicate more coherent topics.

C_V: Coherence Metric C_V combines NPMI with cosine similarity (Röder et al., 2015). For each topic $W_k = \{w_1, \ldots, w_Q\}$, it computes a vector $\mathbf{v}_{\text{NPMI}}(w_i) = \{\text{NPMI}(w_i, w_j)\}_{j=1,\ldots,Q}$ for each word w_i. It then averages the cosine similarity between each word's NPMI vector and a context vector representing the aggregated NPMI scores for all words in the topic. Specifically:

$$C_V(W_k) = \frac{1}{Q} \sum_{i=1}^{Q} \text{sim}(\mathbf{v}_{\text{NPMI}}(w_i), \sum_{j \neq i} \mathbf{v}_{\text{NPMI}}(w_j)) \tag{18}$$

where

$$\mathbf{v}_{\text{NPMI}}(w_i) = \{\text{NPMI}(w_i, w_j)\}_{j=1,\ldots,Q}, \tag{19}$$

$$\mathbf{v}_{\text{NPMI}}(\{w_1, w_2, \ldots, w_Q\}) = \Big\{ \sum_{i=1}^{Q} \text{NPMI}(w_i, w_j) \Big\}_{j=1,\ldots,Q}. \tag{20}$$

(Wu et al., 2024), and cos is defined as the cosine between two vectors $\mathbf{u}, \mathbf{v} \in \mathbb{R}^N$ as

$$\cos(\theta) = \frac{\mathbf{u} \cdot \mathbf{v}}{\|\mathbf{u}\| \, \|\mathbf{v}\|} \tag{21}$$

where $\mathbf{u} \cdot \mathbf{v}$ denotes the dot product of the vectors, and $\|\mathbf{u}\|$, $\|\mathbf{v}\|$ are their Euclidean norms. The overall C_V is the average over all topics. Higher values are better.

A.3.2 Topic Diversity

Measures distinctiveness between different topics.

D_{PJD}: Pairwise Jaccard Distance Computes the average Jaccard distance between all pairs of topic keyword sets W_i, W_j (Tran et al., 2013).

$$J(W_i, W_j) = 1 - \frac{|W_i \cap W_j|}{|W_i \cup W_j|}. \tag{22}$$

D_{PJD} is the average of $J(W_i, W_j)$ over all unique pairs of topics. A score closer to 1 indicates higher diversity (less overlap).

D_{PUW}: Proportion of Unique Words Measures the percentage of unique words across all top-Q words of all K topics (Dieng et al., 2019).

$$D_{PUW} = \frac{\left| \bigcup_{k=1}^{K} W_k \right|}{K \cdot Q}. \tag{23}$$

A score of 1 means all keywords across all topics are unique.

A.3.3 Topic Validity

Assesses the practical usefulness of the topics.

U_{LLM}: LLM-based Usefulness Evaluation An LLM is prompted to evaluate the generated topics based on a user-defined purpose (provided to Textonomy). The LLM assesses the topic set on criteria including: Relevance, Clarity, Comprehensiveness, and Distribution. Each criterion is scored [0,1], and U_{LLM} is the average score, aiming to capture alignment with user goals (Hoyle et al., 2022).

U_{OR}: Outlier Ratio Measures the proportion of documents not assigned to any topic (outliers).

$$U_{OR} = \frac{|D_{\text{out}}|}{N} \tag{24}$$

where $D_{\text{out}} = \{d_j \in D \mid T(d_j) = \emptyset \text{ or is an outlier topic}\}$. A lower U_{OR} is generally preferred for tasks requiring comprehensive categorization.

B Reproducibility and Hyperparameters

To facilitate reproducibility, we detail our settings in Table 5 for the conducted experiments. The full configurations, code, prompts, and data are also made available in the accompanying files and will be made publicly available as a Github repository upon publication.

C Figures for Insights from Qualitative Comparison to Ground Truth

Here is Figure 4 and Figure 5 that are referred to in Subsection 5.5.

Parameter	Textonomy (Proprietary)		Textonomy (Open Source)	
	Wiki Dataset	**Bills Dataset**	**Wiki Dataset**	**Bills Dataset**
		Data & Sampling		
Dataset Name	`wikitext-103`	`bills`	`wikitext-103`	`bills`
Taxonomy Sample Size	500	1,100	500	500
Classifier Sample Size	1,350	1,340	1,340	1,340
		LLM Models & Behavior		
Summarization LLM	`gpt-4o-mini-2024-07-18`	`gpt-4o-mini-2024-07-18`	`mistral-large-2411`	`mistral-large-2411`
Taxonomy LLM	`o3-mini-2025-01-31`	`o3-mini-2025-01-31`	`mistral-large-2411`	`mistral-large-2411`
Pseudo-Labeling LLM	`gpt-4o-mini-2024-07-18`	`gpt-4o-mini-2024-07-18`	`mistral-large-2411`	`mistral-large-2411`
Temperature	0.1	0.1	0.1	0.1
API Call Retries	5	5	15	15
API Wait Time (s)	10	10	2	2
Max Summarization Threads	20	20	1	1
		Content Generation		
Use Case Prompt	A taxonomy to organise these articles into main categories. Aim at around 10-20 categories. Examples: 'Music', 'Social sciences and society'...	A taxonomy to organise these bills into categories by topics. Aim at around 20-25 topics. Examples: 'Government Operations', 'Environment'...	A taxonomy to organise these articles into main categories. Aim at around 10-20 categories. Examples: 'Music', 'Social sciences and society'...	A taxonomy to organise these summaries of US congressional bills into (main) topics. Aim at around 15-25 topics. Examples: 'Government Operations', 'Environment'...
Max # of Topics (k)	25	55	25	25
Topic Name Length	4	4	4	4
Topic Description Length	15	15	20	20
Summary Length	30	30	50	50
Explanations Length	15	15	20	20
Taxonomy Refinement Iter.	4	10	4	4
		Classifier		
Encoder Model	`all-MiniLM-L6-v2`	`all-MiniLM-L6-v2`	`all-MiniLM-L6-v2`	`all-MiniLM-L6-v2`
Train/Test Split Ratio	0.75	0.75	0.75	0.75

Table 5: Hyperparameter configurations for the main experiments. Lengths are specified in words.

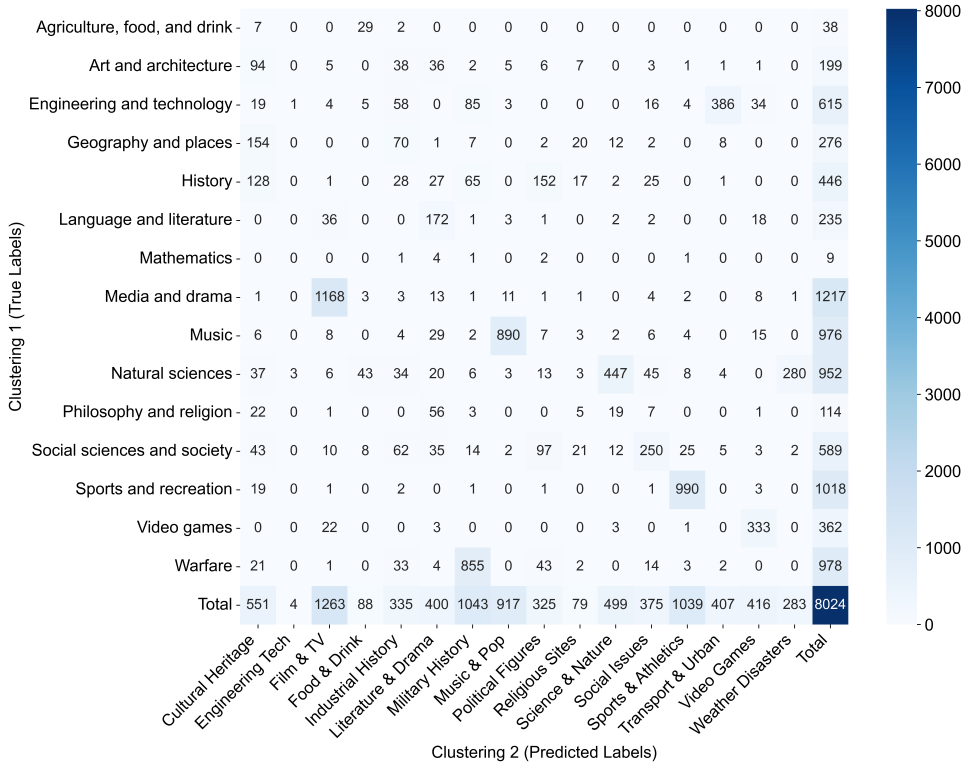

Figure 4: The contingency matrix of the ground-truth clustering and the best topical alignment experiment run of Textonomy.

Ground truth:

Media and drama
Language and literature
Music
Video games
Art and architecture
Philosophy and religion
Agriculture, food, and drink
Warfare
Engineering and technology
History
Social sciences and society
Sports and recreation
Natural sciences
Mathematics
Geography and places

Textonomy:

Film & TV: Movies, television episodes, and cinematic productions.
Literature & Drama: Classic literature, theater, and dramatic cultural works.
Music & Pop: Popular songs, albums, and music celebrity content.
Video Games: Interactive game reviews, summaries, and gaming culture.
Cultural Heritage: Historic sites, traditional arts, and architectural legacies.
Religious Sites: Churches, temples, and sacred architecture of importance.
Food & Drink: Culinary arts, recipes, and food production cultural topics.
Military History: Battles, campaigns, and military operations across eras.
Engineering Tech: Innovative engineering projects, technical designs, and solutions.
Political Figures: Key political leaders and influential governmental actors.
Industrial History: Evolution of manufacturing, industry, and economic heritage.
Social Issues: Controversies, legal cases, and crime-related narratives.
Sports & Athletics: Athletic events, sports figures, and competitive achievements.
Science & Nature: Scientific discoveries, mathematical theories, and natural phenomena.
Transport & Urban: Highways, railways, and urban planning infrastructure.
Weather Disasters: Storms, hurricanes, and notable natural disaster events.

Figure 5: A comparison of the set of ground-truth labels with the taxonomy produced by the best topical alignment experiment run of Textonomy. The color coding represents supposed agreement or at least a partial overlap between the two sets of labels. Text without clear agreement is colored black.